

إعداد

المهندس عدنان شاهين

مراجعة وتدقيق

آلاء المتني - يمان رجب

Designed by: 3Ns Designs

بسم الله الرحمن الرحيم

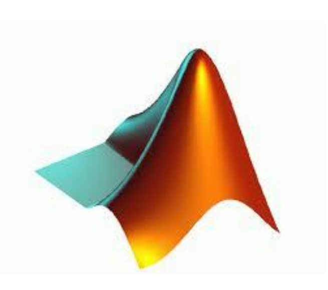
وَقُلِ اعْمَلُوا فَسَيَرَى اللَّهُ عَمَلَكُمْ وَرَسُولُهُ وَالْمُؤْمِنُونَ

صدق الله العظيم

التوبة (105)

شكر و تقدير :

أوجه جزيل الشكر للمهندسة "آلاء المتني" والمهندسة "يمان رجب" و جميع الذين استمتعت بالعمل معهم. أشكر كذلك المهندس "نور الدين الهبل" الذي قام بكافة تصاميم الغرافيك لهذا الكتاب.



ثلاثي النجاح :

- جعل النية في كل الأعمال خالصة لوجه الله .
- إتقان كل عمل نقوم به، انطلاقاً من حديث رسول الله

صلى الله عليه و سلم :

"إن الله يحب إذا عمل أحدكم عملا أن يتقنه " .

- الإرادة القوية التي لا تغيرها صعوبات الحياة.

الفهرس

5. برمجة واجهة المستخدم الرسومية gui
6. برمجة القوائم وأشرطة الأدوات
7. قيادة المعطيات والربط بين عدة واجهات

الجزء الثالث:

التحكم بالمنافذ والربط مع المتحكمات

1. التحكم بالمنفذ التفرعي
2. التحكم بالمنفذ التسلسلي
3. النافذة اللاتزامنية من طرف المتحكمات UART
4. أمثلة
5. التحكم بالأردوينو
6. امثلة عملية

الجزء الرابع :

تحليل الدارات الكهربائية

1. تحليل دارات التيار المستمر
2. تحليل دارات التيار المتناوب
3. دراسة الحالات العابرة في الدارات الكهربائية
4. نظرية الحالات المتغيرة
5. رباعيات الأقطاب

الجزء الخامس :

أساسيات المحاكاة في الماتلاب

1. تعريف المكتبة أو الأداة simulink
2. نمذجة ومحاكاة نماذج بسيطة لأنظمة هندسية
   - المبدل التمثيلي - الرقمي
   - ماسك النظام الصفري والأول
   - التشكيلات المختلفة للمرشحات الرقمية
3. تمثيل و محاكاة الدارات الكهربائية
   - النمذجة والمحاكاة لدارة كهربائية تسلسلية
   - النمذجة والمحاكاة لدارة كهربائية تسلسلية – تفرعية
   - النمذجة والمحاكاة لدارة كهربائية مختلطة

الجزء السادس :

نمذجة ومحاكاة الآلات الكهربائية

1. التحويلات ثلاثية الطور
   - تحويل كلارك

مقدمة :

يعتبر الماتلاب MATLAB أحد أهم البرامج التي تتيح بيئة برمجية عالية المستوى بالإضافة لبيئة محاكاة للأنظمة الهندسية المختلفة، وقد تم مؤخراً تطوير كمية كبيرة من الأدوات في هذا البرنامج، فقد تمت إضافة مكاتب خاصة بنظم القدرة و الطاقات المتجددة، وأخرى لنظم الاتصالات وأنظمة التحكم الآلي و الأتمتة الصناعية و الكثير من المكاتب و الأدوات الأخرى المساعدة لمختلف التخصصات الهندسية. و يعتبر هذا البرنامج أيضاً أحد أهم المقومات الأساسية للدراسة الأكاديمية والبحث العلمي، و قد تم اعتماده ليُدرس كمادة أساسية لبناء التفكير البرمجي السليم للمهندس في كثير من الجامعات.

ولذلك ونظراً لكل ما سبق ذكره فقد عملت لساعات طويلة في هذا الكتاب على وضع شرح مبسط لأهم المواضيع التي تهم المهندس بدءاً بأساسيات البرمجة في الماتلاب انتهاءً بتطبيقات متقدمة المستوى في مجال محاكاة الآلات الكهربائية أحادية و ثلاثية الطور. وأضع بين أيديكم الآن هذا الكتاب سائلاً الله أن أكون قد وفقت في كتابة أسطره.

محتوى الكتاب :

الجزء الأول : أساسيات البرمجة في الماتلاب ، في هذا الجزء من الكتاب يتم شرح المواضيع الأساسية للبرمجة والتي لابد لأي شخص يرغب باستخدام البيئة البرمجية في الماتلاب أو بيئة المحاكاة بأن يكون محيط بها و قادر على التعامل معها بمرونة. تم دعم هذا الجزء بعدد كبير من الأمثلة التوضيحية التي تسمح للمبتدأ في البرمجة بتعلم هذه الأساسيات بسهولة و التعامل معها بمرونة. يدور الحديث في هذا الجزء عن المصفوفات، بنى التحكم، التوابع، عمليات الإدخال و الإخراج بالإضافة للرسوميات و التحليل الرياضي و كيفية حل المعادلات التفاضلية و التعامل مع كثيرات الحدود و المزيد ..

الجزء الثاني : واجهات المستخدم الرسومية ، في هذا الجزء تم وضع شرح تفصيلي لبرمجة واجهات المستخدم الرسومية باستخدام الأداة GUIDE حيث تم شرح كل ما يتعلق بخصائص هذه الأداة و ميزاتها وكيفية الاستفادة منها بأكبر قدر. وتم تسليط الضوء على بعض الأمور المتقدمة كالربط بين الواجهات و برمجة القوائم و أشرطة الأدوات. وضعت بعض الأمثلة البسيطة في هذا الجزء من الكتاب تاركاً للقارئ وظيفة التعمق في التنفيذ العملي في مزيد من التطبيقات المختلفة.

الجزء الثالث : التحكم بالمنافذ و الربط مع المتحكمات ، نتعلم هنا كيفية التحكم بالنافذة التسلسلية و النافذة التفرعية، وكيفية إعداد بروتوكول الاتصال بين طرفيتين، كيفية الربط مع المتحكمات الصغرية وإجراء عملية المحاكاة برمجياً. كما يتم في نهاية هذا الجزء شرح كيفية الربط مع الأردوينو و إنشاء بروتوكول اتصال وتجهيز الدارة للتحكم المباشر من البيئة البرمجية في الماتلاب.

الجزء الرابع : تحليل الدارات الكهربائية ، يعتبر هذا الجزء عبارة عن مجموعة من التطبيقات الهندسية لأساسيات البرمجة في الماتلاب. في هذا الجزء نتعلم كيف يمكن تحليل دارات التيار المستمر والتيار المتناوب و كذلك دراسة و تحليل الأنظمة ثلاثية الطور، وننهي هذا الجزء بتحليل و دراسة الدارات الكهربائية خلال الحالة العابرة. تم تسليط الضوء على نظرية الحالات المتغيرة (State Variable) لأهميتها في المراحل المتقدمة من الدراسة الأكاديمية.

الجزء الخامس : أساسيات المحاكاة في الماتلاب ، نبدأ في هذا الجزء بشرح أساسيات النمذجة و المحاكاة في الماتلاب ، نتعرف على أهم المكاتب الموجودة في الماتلاب وكيفية الوصول إليها، كيف يمكن إنشاء نظام محاكاة باستخدام هذه البيئة. و بعد ذلك ننتقل لمحاكاة بعض الأنظمة الهندسية المختلفة مثل أنظمة التحكم و الدارات الكهربائية ....

الجزء السادس : نمذجة و محاكاة الآلات الكهربائية ، هذا الجزء للمتخصصين في دراسة و تحليل الآلات الكهربائية، دراسة الآلة في الحالة العابرة أوالمستقرة و تنفيذ الاختبارات المختلفة عليها لدراسة أداء الآلة. قمت بداية بشرح مبسط عن مبادئ محاكاة الآلات الكهربائية و كيفية تمثيل النظام رياضياً و انتقلت مباشرة للأمثلة العملية كمحاكاة المحولات أحادية الطور والثلاثية الطور وآلات التيار المتناوب المتخلفة.

اسأل الله أن أكون قد وفقت في عملي هذا و اترككم للتمتع في قراءة وتنفيذ أسطر هذا الكتاب.

م. عدنان شاهين



أساسيات البرمجة في الماتلاب

- التعامل مع المصفوفات.
- البرمجة غير المرئية.
- الرسم ثنائي البعد.
- التوابع الرياضية الخاصة في الماتلاب.
- تطبيقات هندسية وأمثلة عملية.

التعامل مع المصفوفات

١,١ مقدمة

تتكون المصفوفة من عدد من الصفوف وعدد من الأعمدة حيث يرمز بمتحول ما لكل منهما وتستخدم المصفوفات في كثير من التطبيقات الهندسية لحل المشاكل المعقدة ، وعناصر المصفوفة ممكن أن تكون قيم حقيقية أو عقدية ، ويرمز عادة للمصفوفة

Matricname(m,n)

حيث : m عدد الأسطر n عدد الأعمدة

٢,١ العمليات الأساسية على المصفوفات

- إنشاء مصفوفة (٤،٤) باسم A

A = [3 4 4 3 ; 4 5 6 7 ; 5 6 7 4 ; 4 5 6 7]

إدخال السطر السابق في سطر الأوامر في ماتلاب يشكل لدي مصفوفة مربعة بالقياس سابق الذكر حيث نلاحظ أنه عندما نريد الانتقال من سطر لسطر لإدخال عناصر مصفوفة نضع فاصلة منقوطة وذلك كما هو موضح ، ويكون شكل المصفوفة ...

```
3  4  4  3
4  5  6  7
5  6  7  4
4  5  6  7
```

- استدعاء عنصر من مصفوفة عن طريق رقم السطر ورقم العمود.

مثلاً لاستدعاء العنصر الثاني من السطر الثالث نكتب :

A(3,2)   ans = 6

- استدعاء عنصر من المصفوفة عن طريق ترتيبه.

A(8)   ans = 5

A(1)   ans = 7

حيث يتم العد حسب الترتيب من أعلى العمود إلى اسفله وبالترتيب.

- استدعاء سطر من مصفوفة.

A(3,:)   ans = 5 6 7 4

- استدعاء عمود من مصفوفة.

A(:,4)

3
7
4
7

- استدعاء السطر الثاني الذي يحوي العناصر الثلاثة الأخيرة منه (دون العنصر الأول).

A(2,2:4)   ans = 5 6 7

- استدعاء العنصر الأول والأخير من السطر الثالث.

A(3,1:3:4)   ans = 5 4

- إضافة عمود في نهاية المصفوفة.

A(:,5) = [3 5 0 0]   فتصبح المصفوفة كما هو مبين

3	4	4	3	3
4	5	6	7	5
5	6	7	4	0
4	5	6	7	0

- إضافة عمود جديد للمصفوفة الجديدة بحيث يكون العنصر الأخير من السطر الثالث مساوي٩

A(3,6) = 9   فتصبح المصفوفة كما هو مبين

3	4	4	3	3	0
4	5	6	7	5	0
5	6	7	4	0	9
4	5	6	7	0	0

- حذف العمود الثاني من المصفوفة الجديدة.

A(:,2) = [ ]   فتصبح المصفوفة كما هو مبين

3	4	3	3	0
4	6	7	5	0
5	7	4	0	9
4	6	7	0	0

- استبدال عنصر من المصفوفة بعنصر جديد.

A(3,4) = 11 فتصبح المصفوفة كما هو مبين

3	4	3	3	0
4	6	7	5	0
5	7	4	11	9
4	6	7	0	0

- استدعاء آخر سطر من المصفوفة.

A(end,:)

ملاحظة :

١- تشكيل مصفوفة واحدية ones(m,n)

2 - تشكيل مصفوفة صفرية zeros(m,n)

٣- تشكيل مصفوفة محايدة eye(m,n)

٣,١ العمليات الرياضية على المصفوفات

بفرض لدينا المصفوفات A,B ونريد إجراء عمليات الجمع والطرح والضرب والقسمة ...إلخ عليها فتكون كالتالي :

١- الجمع A+B
٢- الطرح A-B
٣- الضرب A.*B
٤- القسمة A./B
٥- الرفع لقوى A.^B
٦- الضرب برقم ٣ 3*A

٧- جمع مصفوفة لرقم ٢   A+2   (حيث يتم جمع العدد ٢ لجميع عناصر المصفوفة)

٤,١   عمليات أخرى على المصفوفات

لتكن المصفوفة D : $\begin{matrix} 3 & 4 \\ 2 & 0 \end{matrix}$

- إيجاد منقول مصفوفة (استبدال الأعمدة مع الأسطر)...

T = D′   فتكون المصفوفة الناتجة T

$$\begin{matrix} 3 & 2 \\ 4 & 0 \end{matrix}$$

- إيجاد مقلوب مصفوفة ...   S = inv(D)   فتكون المصفوفة الناتجة S

$$\begin{matrix} 0 & 0.5 \\ 0.25 & -0.375 \end{matrix}$$

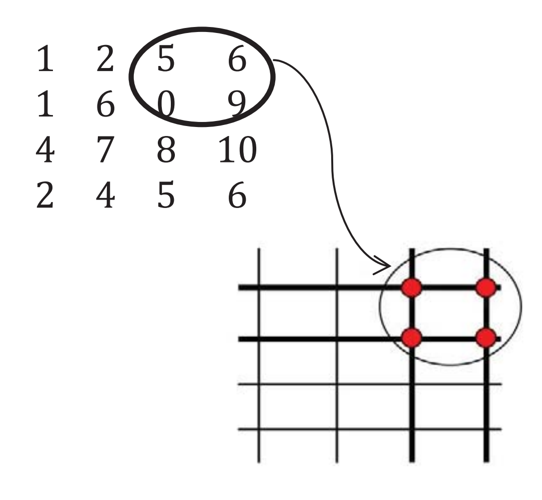
- إيجاد رتبة مصفوفة ...   V = rank(D)

ans = 2

٥,١   عمليات متقدمة على المصفوفات

١,٥,١ استعمال مؤشر المصفوفة مع إحداثيات من عناصر المصفوفة

لتكن المصفوفة C



- استدعاء مصفوفة جزئية من المصفوفة C (كما هو مبين بالشكل)...

c(1:2,3:4) ans =

5	6
0	9

c(3:-1:1,:) ans =

4	7	8	10
1	6	0	9
1	2	5	6

- استبدال ترتيب أعمدة (أو أسطر) المصفوفة مع قيم عناصرها ...

c(:,[3,4,3,4]) ans =

5	6	5	6
0	9	0	9
8	10	8	10
5	6	5	6

c([3,1,3,2],:) ans =

4	7	8	10
1	2	5	6
4	7	8	10
1	6	0	9

٢,٥,١ التعامل مع المصفوفة وتغيير شكلها وتعيين حجمها

لتكن المصفوفة k

1	3	12	10
5	4	3	4
7	0	10	8

- k(:)’

ans = 1 5 7 3 4 0 12 3 10 10 4 8

- إعادة تشكيل المصفوفة بحيث عدد الأسطر (m) وعدد الأعمدة (n) .

reshape (matrix name ,m,n)

مثال :

reshape(k,4,3) ans =

1	4	10
5	0	10
7	12	4
3	3	8

reshape(k,2,6)

ans =	1	7	4	12	10	4
	5	3	0	3	10	8

- تدوير مصفوفة ...

rot90(matrix name)

مثال :

rot90(k)

	10	4	8
	12	3	10
ans =	3	4	0
	1	5	7

- تعيين حجم مصفوفة ...

size (matrix name)

length(matrix name)

مثال :

size(k) ans = 3 4

length(k) ans = 4

٣,٥,١ أنواع من ضرب المصفوفات

لتكن المصفوفتان a السطرية وb العمودية

a = 1 2 3

b =
1
2
3

- الجداء الداخلي أو Dotproduct

a*b ans = 14

حيث تم ضرب العمود الأول مع السطر الأول والعمود الثاني مع السطر الثاني وهكذا ...

- الجداء الخارجي أو outproduct

b*a

	1	2	3
ans =	2	4	6
	3	6	9

حيث تم ضرب عناصر كل سطر من b مع سطر a ...

ملاحظة:

لجعل العملية عملية ترتيبية أي معاملة كل عنصر بالعنصر المناظر له فقط ، يجب وضع نقطة قبل رمز العملية المراد إجراؤها سواء كانت قسمة أو ضرب أو رفع لقوة.

لتكن المصفوفتان : $b = \begin{matrix} 3 & 5 \\ 2 & 1 \end{matrix}$ $a = \begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix}$

- الجداء

a.*b $ans = \begin{matrix} 3 & 10 \\ 6 & 4 \end{matrix}$

- القسمة

a./b $ans = \begin{matrix} 0.3333 & 0.4000 \\ 1.5000 & 4.0000 \end{matrix}$

- الرفع لقوة

a.^b $ans = \begin{matrix} 1 & 32 \\ 9 & 4 \end{matrix}$

ملاحظات:

- إنشاء مصفوفة متزايدة خطياً معلوم عدد عناصرها ...

linspace(a,b,c)

حيث : a القيمة البدائية b القيمة النهائية c عدد عناصر المصفوفة

- إنشاء مصفوفة متزايدة لوغاريتمياً معلوم عدد عناصرها ...

logspace(a,b,c)

حيث : a القيمة البدائية b القيمة النهائية c عدد عناصر المصفوفة

- إنشاء مصفوفة سحرية ...

magic(d) حيث d أبعاد المصفوفة

مثال:

```
v = magic(3)                  8   1   6
                          v = 3   5   7
                              4   9   2
```

وللتحقق من هذه المصفوفة نتأكد من :

١ – مجموع عناصر كل عمود :

```
sum(v)          ans = 15   15   15
```

2 – مجموع عناصر كل سطر :

```
                      15
sum(v,2)        ans = 15
                      15
```

3 – مجموع عناصر القطر الرئيسي :

```
                      8
diag(v)         ans = 5
                      2
```

هذه هي عناصر القطر للمصفوفة السابقة ولإيجاد مجموعها : ans = 15   sum(ans)

- تحدثنا في الدرس عن طريقة استدعاء عنصر من مصفوفة عن طريق ترتيبه حيث أن ترتيب عناصر المصفوفة يكون كالتالي :

```
1   4   7
2   5   8
3   6   9
```

البرمجة غير المرئية

١,٢ مقدمة

سنتعرض فيما يلي عبارات التحكم بمسار البرنامج في الماتلاب والتي تشبه لحد كبير عبارات التحكم في لغات البرمجة الأخرى ، وسندرس :

- الاختبار الشرطي if .
- الاختبار switch .
- حلقة while .
- حلقة For .
- عبارة الإيقاف Break .
- متابعة الحلقة Continue .
- العبارة try .

٢,٢ الحلقات

١,٢,٢ أمر التحكم (الاختبار) الشرطي If : الشكل العام

```
if logical expression
   statements
elseif logical expression
   statements
else
   statements
end
```

عبارة else لا تحتوي على شرط منطقي ولكن يجري تنفيذ العبارات المرتبطة بعبارة else إذا كانت نتيجة الشرط المنطقي في عبارة if (أو elseif) خطأ أو مساوي للصفر.

عبارة elseif تحتوي على شرط منطقي يتم تنفيذه إذا كان الشرط المنطقي عبارة if خطأ أو تساوي الصفر ويتم تنفيذ جميع العبارات المرتبطة بها إذا كانت نتيجة الشرط المنطقي في عبارة elseif تساوي الواحد (لا تساوي الصفر).

٢,٢,٢ الاختبار switch : الشكل العام

```
Switch expression
case value1
   statements
case value2
   statements
otherwise
   statements
end
```

تنفذ عبارة switch مجموعة من العبارات عن قيم معينة للمتحول المراد اختباره .

تتألف العبارة من كلمة switch متبوعة بالعلاقة المراد اختبارها ، وتنفذ العلاقة وتقارن النتيجة مع القيم التي تتبع عبارة case وعند حدوث أول تطابق بين نتيجة العلاقة وبين القيمة الموجودة على يمين عبارة case يتم تنفيذ التي تطابقت القيمة الموجودة على يمينها مع نتيجة العلاقة ، حيث أن الماتلاب لا يتابع المقارنة بعد حدوث اول تطابق.

في حال لم يحدث تطابق مع القيم الموجودة على يمين عبارة case ونتيجة العلاقة المراد اختيارها يتم تنفيذ العبارات التي تلي otherwise وعبارة otherwise اختيارية ، ويجب إنهاء عبارة switch ب end.

٣,٢,٢ حلقة while : الشكل العام

```
while expression
    statements

end
```

تنفذ حلقة while عبارة أو مجموعة من العبارات بشكل متكرر طالما أن قيمة عبارة التحكم تساوي الواحد (أي عبارة التحكم أو الشرط محقق) و إذا كانت عبارة التحكم مصفوفة فيجب أن تحقق جميع عناصر المصفوفة الشرط ليستمر الحساب.

٤,٢,٢ حلقة For : الشكل العام

```
for variable = expression
    statements
end
```

تقوم حلقة for بتكرار عبارات معينة لعدد محدد من المرات .

٥,٢,٢ عبارة الإيقاف break :

تقوم هذه العبارة بإيقاف تنفيذ حلقة For أو حلقة while عند وضعها ضمن الحلقة ... وتتوضح هذه العبارة أكثر بالأمثلة.

٦,٢,٢ عبارة المتابعة Continue :

تقوم هذه العبارة بوقف التكرار الحالي للحلقة ويبدأ في التكرار التالي له بمعنى أن الحلقة تستمر بالعمل لكن عند تعرضها لهذه التعليمة ستقوم بتجاهل التعليمات التي تحتها وتعود لمتابعة الحلقة ... وتتوضح هذه العبارة اكثر بالأمثلة.

٧,٢,٢ العبارة try : الشكل العام

try statement, ... , statement, **catch** statement, ... , statement, **end**

يتم بشكل طبيعي تنفيذ العبارات بين try و catch ولكن عند حدوث أي خطأ في تنفيذ أي عبارة من العبارات عند ذلك يتم الانتقال إلى تنفيذ العبارات الواقعة بين catch و end وعند حصول خطأ أيضاً من أحد العبارات يوقف الماتلاب تنفيذ الأوامر ويضع عبارة الخطأ في متحول اسمه lasterr.

ملاحظات :

١) العبارة return تنهي تسلسل تنفيذ الأوامر في الإجراء وتحول القيادة إلى البرنامج الرئيسي وبشكل عام عندما تنتهي أوامر الإجراء فإن القيادة بشكل آلي تنتقل إلى البرنامج الرئيسي ولكن يمكن وضع هذه التعليمة في أي مكان من الإجراء لإعادة القيادة قسرياً إلى البرنامج الرئيسي.

٢) إظهار رسالة الخطأ على الشاشة يتم بواسطة تابع الخطأ بالشكل :

error('error_message')

حيث عند استدعاء رسالة الخطأ من قبل البرنامج فإن البرنامج يظهر العبارة الموضحة بين إشارات التنصيص.

٣) يمكن إدخال المعلومات إلى البرنامج أثناء تنفيذ ملف m وذلك باستخدام التابع input بالشكل :

n = input('Prompt_string')

حيث يظهر على الشاشة سلسلة الحروف بين إشارتي التنصيص.

٤) يمكن التوقف أثناء تنفيذ البرنامج لمراجعة نتائج الحسابات أو لفحص المخططات ولتحقيق ذلك نستعمل الأوامر

Pause بدون مضمون هذا يسبب توقف البرنامج حتى يضغط المستخدم أي زر من لوحة المفاتيح.

Pause(n) هذا يسبب توقف البرنامج لمدة n ثانية .

٥) عند كتابة أي برنامج يمكن أن تكون طريقة البرمجة بطريقتين وهما إما أن يكتب البرنامج بشكل مباشر وهذه الطريقة تسمى (script) أو أن ننشئ تابع ونضع ضمنه التعليمات البرمجية اللازمة لتنفيذ البرنامج وهذه الطريقة (function) وهنا عدة فروق بين هاتين الطريقتين نذكر منها :

١- الـ function يبدأ بكلمة function بينما الـ script يبدأ بالبرنامج مباشرة.

٢- الشكل الافتراضي للكتابة بعد كلمة function هو أن تضع وسطاء الخرج ومن ثم اسم الـ function وبعدها وسطاء الدخل بينما الـ scripts لا يحتاج إلى كل هذا.

٣- يمكنك التابع (function) عند تنفيذه من تغيير وسطاء الدخل بينما لا يكون هذا ممكن في الـ scripts.

٤- المتغيرات التي تحسب في داخل الـ function لا تحفظ في الـ workspace بينما في الـ scripts تحفظ في الـ workspace (أي في لوحة الدخل والخرج الرئيسية لبرنامج الماتلاب).

كما أننا عندما نريد البدء بكتابة البرنامج لابد في أن نقوم بفتح ملف من نوع M-file من أجل أن نكتب البرنامج فيه بدلاً من الكتابة في الـworkspace لبرنامج الماتلاب وهناك عدة طرق للوصول إلى محرر الـM-file وهي إما من اعلى الشاشة نختار file → new → M-file أو أن تكتب edit في الـworkspace أو أن نضغط على الملف النصي الظاهر في الصفحة الرئيسية للماتلاب ، وعند الإنتهاء من كتابة البرنامج في هذا الملف بقوم بحفظه ومن ثم نقوم بتشغيله من الزر المبين باللون الأخضر اعلى الصفحة ومقاطع الفيديو توضح ذلك أكثر.

٣,٢ أمثلة عملية

- **المثال الأول**

```
clear
n=5
m=6
for i=1:n
   for j=1:m
      i;
      j;
      c(i,j)=i^j
   end
end
```

تم وضع clear في بداية البرنامج لحذف جميع المتغيرات الموجودة في الworkspace وفي هذا البرنامج تكمن وظيفة وضع حلقتي for متداخلتين في توليد مصفوفة ثنائية البعد حيث تكمن وظيفة حلقة for الأولى في توليد أسطر المصفوفة أما حلقة for الثانية فهي لتوليد أعمدة المصفوفة ، حيث هذا البرنامج يقوم بحساب قيمة العنصر من المصفوفة بحيث يساوي رقم السطر الموجود به مرفوعاً بالأس إلى رقم العمود الموجود به هذا العنصر ، فلو كان السطر رقمه ٣ والعمود رقمه ٤ فنلاحظ أن قيمة العنصر ستكون : ٨١ والمصفوفة الناتجة تكون :

```
1    1    1     1     1      1
2    4    8    16    32     64
3    9   27    81   243    729
4   16   64   256  1024   4096
5   25  125   625  3125  15625
```

- **المثال الثاني**

```
clear
n=-6;
if n<0
disp('input must be positive')
elseif rem(n,2)==0
disp('input is even')
else
disp('input is odd')
end
```

هذا البرنامج يسمح بإدخال رقم n له قيمة موجبة حصراً ليميز هذا العدد إن كان فردياً أم زوجياً وفي حال إدخال رقم سالب فإنه يعطي العبارة input must be positive ويكون خرج هذا البرنامج في ال workspace.

اعتمدنا في هذا البرنامج على التابع rem(n,2) حيث يقوم هذا التابع بتقسيم العدد n على ٢ ففي حال وجود باقي للقسمة فلا يتحقق شرط المساواة .

- **المثال الثالث**

```
clear
x = 8
units = 'mm'
switch units
   case {'inch' 'in'}
      x = x*2.54
   case {'feet' 'ft'}
  x=x*12/2.54
   case {'millimeter' 'mm'}
      x=x*0.1
   case {'centimeter' 'cm'}
      x=x
   otherwise
      disp(['هذه الواحدة غير معروفة'])
      x=nan
end
```

يقوم هذا البرنامج بتحويل الرقم من الذي نكتبه من الواحدة المعطاة (المحددة في السطر الثاني للبرنامج) إلى cm ، حيث نلاحظ في المثال الموضح بالبرنامج يكون خرج البرنامج x=0.8.

- **المثال الرابع**

```
for i=1:1000
   elem = input('==>');
   if isempty(elem)
      break
   end
   x(i)=elem
end
x
```

إن التعليمة input تقوم بطباعة مابين إشارتي التنصيص وتوقف عمل البرنامج حتى يدخل المستخدم قيمة ما ويضغط على الزر Enter ، حيث هذا البرنامج يطلب إدخال قيم شعاع قيمة قيمة حتى يدخل المستخدم قيمة فارغة لتقوم break بإنهاء البرنامج.

- **المثال الخامس**

```
function vals1(a,b,c)
delta = b^2-4*a*c
if delta>0
   x1=(-b+sqrt(delta))/(2*a)
```

```
    x2=(-b-sqrt(delta))/(2*a)
elseif delta<0
    disp('the roots are complex')
else
    x1_2=(-b/(2*a))
end
```

برنامج حل معادلة من الدرجة الثانية بشرط أن تكون حلول هذه المعادلة لا تحوي أي عدد عقدي ، نلاحظ أننا انشأنا تابع لحل المعادلة.

إن اسم التابع vals1 هو اختياري للمستخدم ويمكن له تغييره كما يشاء ولكن يجب ملاحظة انه عند حفظ البرنامج يجب حفظه بنفس اسم التابع ومابين قوسين هي الدخل أي القيم التي سيقوم المستخدم بإدخالها ، ولإستدعاء هذا التابع نكتب في الworkspace

vals1(4,6,2)

فيعطي

Delta = 4   x1=-0.5   x2=-1

أو بطريقة ثانية نكتب :

a=6   b=8   c=3

vals1(a,b,c)

delta = -8

the roots are complex

- **المثال السادس**

```
function max1(a,b,c)
if a>b
    max=a;
    if c>max
        max=c;
    end
else
    max=b;
    if c>max
        max=c;
    end
end
max
```

يقوم هذا البرنامج بتحديد القيمة الأكبر من بين ثلاثة أعداد يدخلها المستخدم ، حيث يتم استدعاء التابع الخاص بهذا البرنامج كما يتم الاستدعاء في البرنامج السابق وبإحدى الطريقتين.

- ## المثال السابع

```
function c=factor1(n)
v=1
for i=1:n
   v=v*i;
end
v
```

يقوم هذا البرنامج بحساب قيمة العاملي لرقم ما يحدده المستخدم ، حيث انشأنا تابع كما في الأمثلة السابقة وطريقة الاستدعاء أيضاً كما ذكرنا في الجلسة السابقة.

- ## المثال الثامن

```
function g=sort1(a)
s=length(a);
for i=1:s-1
   for j=i+1:s
      if a(i)<a(j)
         x=a(i);
         a(i)=a(j);
         a(j)=x;
      end
   end
end
a
```

يقوم هذا البرنامج بترتيب عناصر مصفوفة (شعاعية) أي مكونة من سطر واحد ترتيباً تنازلياً ، حيث يقارن هذاالتابع قيمة كل عنصر من المصفوفة مع قيمةالعنصر التالي وبناءً على هذا الترتيب يقوم بترتيب عناصر من المصفوفة من جديد.

- ## المثال التاسع

```
function prod22(num)
if length(num)~=1|~isnumeric(num)
   disp('please enter one number')
else
   for i=1:10
      disp([num2str(num) 'x' num2str(i) '=' num2str(num*i)])
   end
end
```

يقوم هذا البرنامج بإظهار جدول الضرب لأي رقم يحدده المستخدم ، حيث في السطر الثاني من البرنامج يقوم البرنامج بفحص القيمة المدخلة ويتأكد أن لا تكون لا محرف ولا أكثر من رقم ، حيث

نقوم بتحويل الأرقام إلى محارف بالتعليمة num2str أي number to string وبعد القيام بالعملية المطلوبة بعيد هذه القيمة الرقمية لمحارف بتعليمة معاكسة للسابقة num2str.

• المثال العاشر

```
function calcul2(a,b,operate)
if operate=='+' | operate=='*' | operate=='-' | operarte=='/'
   eval([num2str(a) operate num2str(b)])
else
   disp('enter one of this operation{+,-,*,/}')
end
```

برنامج آلة حاسبة بسيطة حيث يقوم المستخدم بإدخال الرقم الأول والثاني ومن ثم العملية المطلوبة فيعطي البرنامج ناتج العملية بين الرقمين ، وفي حال ادخل المستخدم عملية غير معروفة يعطي البرنامج العبارة

'enter one of this operation{+,-,*,/}

في هذا البرنامج نلاحظ أننا استخدمنا التابع eval وتكمن وظيفة هذا التابع في الماتلاب في تنفيذ العبارات المحرفية التي لها معنى في الماتلاب ، فلو كتبنا في السطر الثالث للبرنامج العبارة :

z=([num2str(a) operate num2str(b)])

بدلاً من العبارة السابقة أي دون استخدام التابع eval نلاحظ أن البرنامج سيكون خرجه (a+b) "في حال الجمع ومهما كانت قيم a و b"

• المثال الحادي عشر

```
function game4(c)
n=randperm(c);
for i=1:inf
   f=input('enter the number please:')
   if f<n(4)
      disp('the number is greater than your number')
   elseif f>n(4)
      disp('the number is smaller than your number')
   else
      disp('that is right')
      disp('congratulaion!!!')
      disp(['there are ' num2str(i) 'attempts'])
      break
   end
end
```

هذا البرنامج عبارة عن لعبة مسلية يقوم فيها البرنامج بتخزين رقم ويطلب من اللاعب إدخال قيمة للرقم الموجود في ذاكرة البرنامج ليخبره إن كان الرقم أصغر أو أكبر أو يساوي الرقم المخزن في

الذاكرة ، وبعد إدخال الرقم الصحيح يعطي البرنامج عدد المحاولات التي قام بها اللاعب حتى توصل للرقم الصحيح.
تعليمة randperm تقوم بتوليد أرقام وبترتيب عشوائي والرقم الأعظمي لهذه الأرقام يحدد من قبل المستخدم ويرمز له في برنامجنا هذا c.

الرسم ثنائي البعد

١,٣ مقدمة

سنتعرض في هذه الجلسة لدراسة أهم الأوامر التي تتعلق بالرسم ثنائي البعد في الماتلاب وطرق المعالجة والتنسيق للرسوم البيانية.

٢,٣ الخصائص الرئيسية للرسم البياني ثنائي البعد في الماتلاب

١,٢,٣ تعليمة الرسم الأساسية :

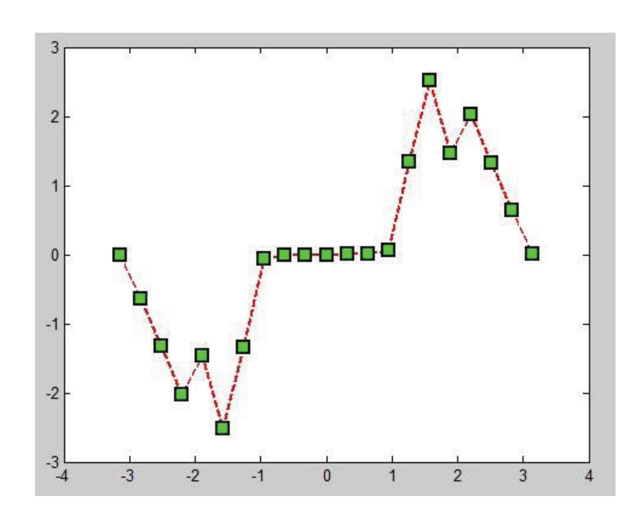
```
plot(Y)
plot(X1,Y1,....)
plot(X1,Y1,LineSpec,....)
plot(...,'PropertyName',PropertyName,...)
plot(axes_handle,...)
h = plot(...)
hlines = plot('v6',...)
```

ومن الخصائص الإضافية لهذه التعليمة أيضاً :

- LineWidth : تحديد عرض الخط (سماكة)
- MarkerEdgeColor : تحديد لون حواف العلامات المميزة
- MarkerFaceColor : تحديد لون العلامات المميزة
- MarkerSize : تحديد قياس العلامة المميزة

مثال :

```
x=-pi:pi/10:pi;
y=tan(sin(x))-sin(tan(x));
plot(x,y,'--
rs','linewidth',2,'markeredgecolor','k','markerfacecolor','g','marke
rsize',10);
```



ملاحظة : إعطاء قيم متقاربة للمتغير x يجعل دقة الرسم أعلى .
الجداول التالية توضح الرموز الدالة على الألوان والرموز الدالة على خطوط الرسم والرموز الدالة على نماذج العلام والمستخدمة في تعليمة الرسم جسم التابع plot والتي سنراها في أمثلة لاحقة.

الرمز	اللون	
c	Cyan	أخضر مزرق
m	Magenta	ارجواني
y	Yellow	أصفر
k	Black	أسود
b	Blue	أزرق
r	Red	أحمر
G	Green	أخضر

نموذج خط الرسم	الرمز
________________________	" _ "
--------------------------	" -- "
بدون خط رسم	None
....	" : "
-.-.-.-.-.-.-.-.-.-	".-"

الدلالة	الرمز
إشارة جمع	+
دائرة	O
نجمة	*

.	نقطة
>	مثلث رأسه لليمين
<	مثلث رأسه لليسار
^	مثلث رأسه للأعلى

وهناك أيضاً نماذج أخرى يمكن الإطلاع عليها من خلال أمر المساعدة إذا دعت الحاجة لها.

٢,٢,٣ رسم منحنيين على شكل واحد :

مثال (١) :

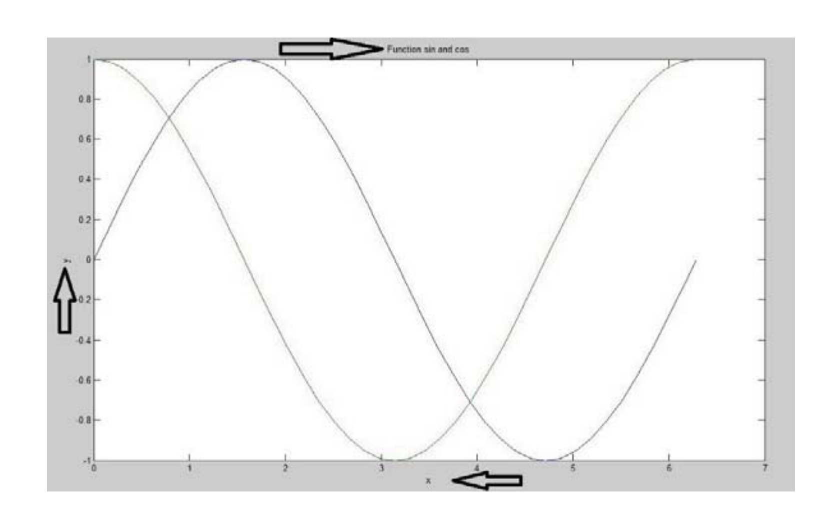
```
x = 0:0.05*pi:2*pi;
y = sin(x)
z = cos(x)
plot(x,y,x,z)
```

وبذلك نستطيع أن رسم عدد كبير من المنحنيات على نفس الشكل.

٣,٢,٣ تسمية الشكل والمحاور :

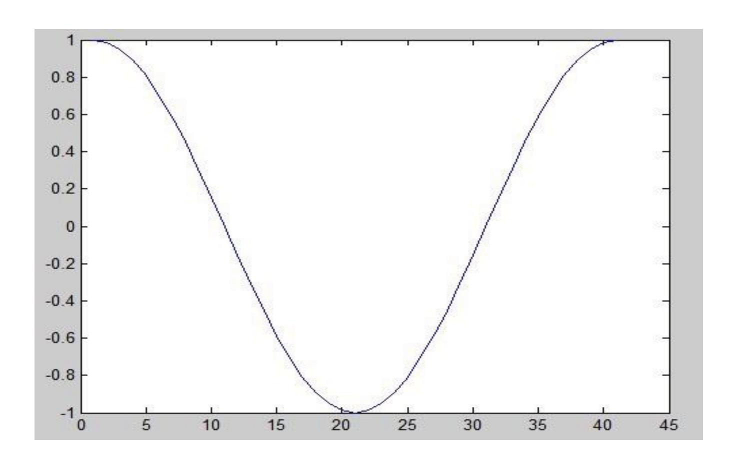
لو أضفنا الأسطر التالية على البرنامج السابق يصبح الرسم ...

```
title('Function sin and cos')
xlabel('x')
ylabel('y')
```



مثال (٢) :

```
z = cos(x)
plot(z)
```



يأخذ التابع أول قيمة من التابع y ويأخذ x رقم العنصر أي كأنك كتبت : plot(1:length(z),z)

أما لو كانت z عقدية فإنه يرسم القسم العقدي بالنسبة للقسم الحقيقي.

٤,٣,٢ تلوين الأشكال :

الجدول الموضح في الأعلى يوضح رموز الألوان التي نريد الرسم فيها.

٥,٣,٢ إنشاء شبكة :

إن الأمر grid on يعطيك شبكة حسب تقسيمات المحاور ولإزالة الشبكة مرة أخرى نكتب grid off .

٦,٣,٢ الكتابة على الشكل :

نستطيع كتابة أي نص على الرسم وذلك بتحديد إحداثياته فقط : text(5,0.1,’sin’)

٧,٣,٢ أبعاد المحاور :

إن البرنامج بشكل تلقائي يحدد لك أبعاد المحاور ولكن لو أردت تحديد هذه الحدود فإليك التعليمة التالية : axis([-15 15 -17 17]) إن أول رقمين يحددان مجال المحور x وثاني رقمين يحددان مجال المحور y.

٨,٣,٢ تسمية كل منحني حسب لونه :

إذا كان لدينا شكل يحوي عدة توابع متداخلة وأردنا تعريف كل تابع حسب لونه أو غيره فيمكن استعمال التعليمة التالية :

legend(‘sin(x)’,’cos(x)’,’sin(x)./x’)

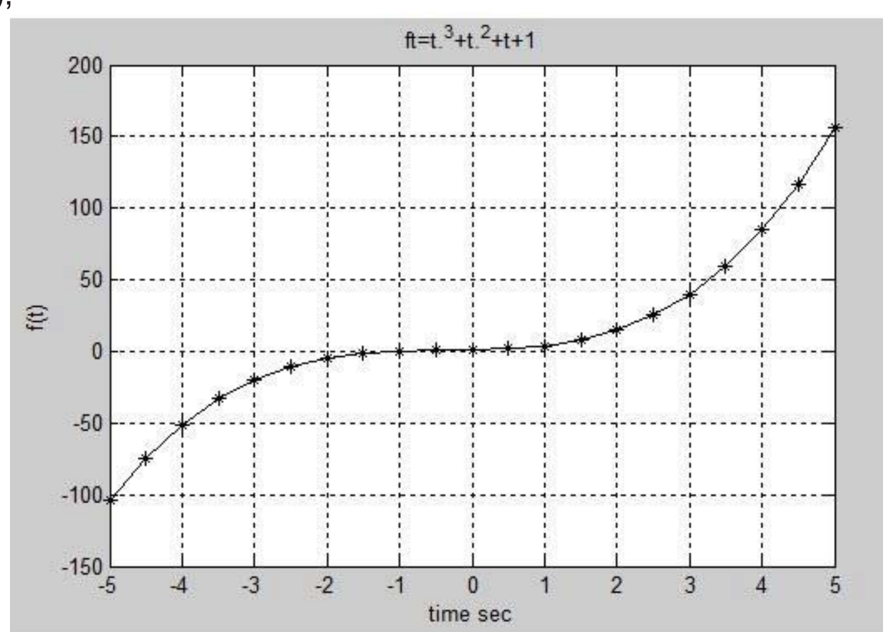
هذا الأمر سيأخذ لون الشكل الأول ويضع بجانبه أول عبارة تضعها في أمر legend والثاني مع الثاني وهكذا ...

٣,٣ أمثلة عامة

- المثال الأول :

برنامج لرسم تغيرات كثير الحدود $f(t) = {}^{3} + t^2 + t^1 + 1$ مع تغيرات t.

```
t=-5:0.5:5;
ft=t.^3+t.^2+t+1;
plot(t,ft,'-*k');
grid on
title('ft=t.^3+t.^2+t+1');
xlabel('time sec');
ylabel('f(t)');
```



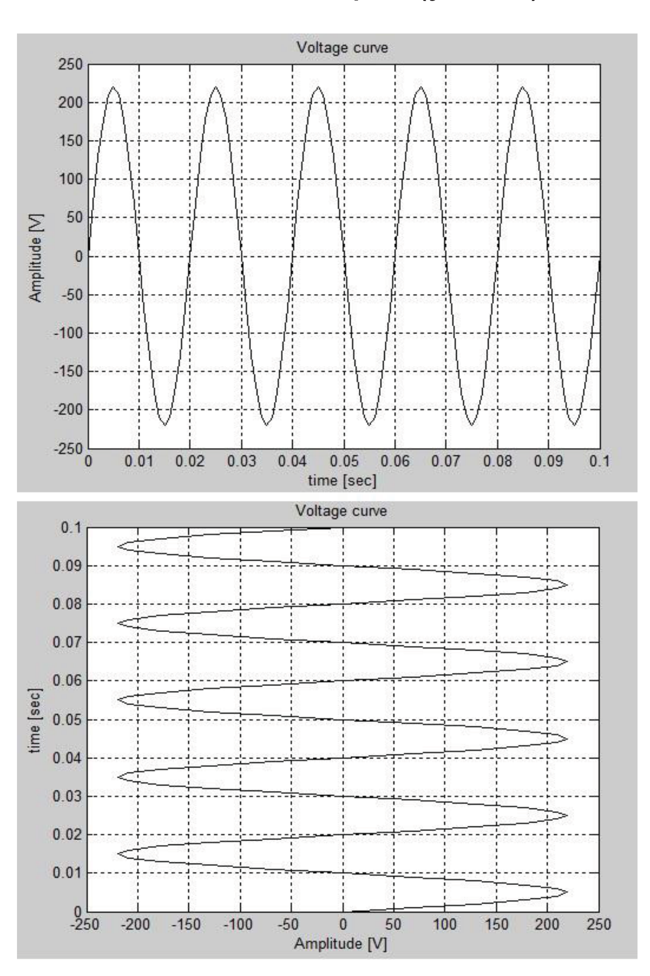
- المثال الثاني :

برنامج لرسم تغيرات التابع البسيط y = A sin(wt) مع تغيرات الزمن.

```
t=0:0.001:0.1;
A=220;
f=50;
w=2*pi*f;
y=A*sin(w*t);
```

```
plot(t,y,'k-')
title('Voltage curve');
xlabel('time [sec]');
ylabel('Amplitude [V]');
grid on;
pause
plot(y,t,'k-')
title('Voltage curve');
xlabel('Amplitude [V]');
ylabel('time [sec]');
grid on;
```

سبق وشرحنا عم التابع pause في درس سابق ، حيث هذا التابع يقوم بإيقاف البرنامج حتى يضغط المستخدم أي زر ليتمم عمل البرنامج حتى نهايته ، في مثالنا هذا يقوم البرنامج بعملية الرسم وفق العلاقة plot(t,y,’k-‘) وعندما يضغط المستخدم أي زر من لوحة المفاتيح فإن البرنامج سوف يحذف الرسم ليرسم وفقاً للعلاقة plot(y,t,’k-‘).



سأكتفي بهذا القدر بالنسبة للرسم ثنائي البعد في الماتلاب وسأدع الأمور الأخرى لك كي تتعلمها لوحدك وذلك بالاستعانة ببيئة الـ help المتوفرة في الماتلاب والتي تعد طريقة ممتازة للبحث تعلم الماتلاب ، علماً ان برنامج الماتلاب يوفر إمكانية الرسم الثنائي البعد بالصيغة القطبية وبالصيغة العقدية ويوفر إمكانية الرسم على شكل أعمدة من أجل التطبيقات الإحصائية أو من أجل إظهار قيم التوافقيات في التوترات والتيارات في التطبيقات الهندسية ، كما يمكننا البرنامج من الرسم على لوحة بأبعاد الرسم اللوغاريتمية.

التوابع الرياضية الخاصة في الماتلاب

١,٤ مقدمة

يتم التعامل مع كثيرات الحدود في الماتلاب من خلال توابع خاصة أعدت لحل كثيرات الحدود ضمن هذا البرنامج ، حيث يتم تحويل كثير الحدود إلى شعاع سطري ، وتمثل الأعداد ضمن هذا النسق معاملات كثير الحدود ، وترتب ضمن النسق بشكل تنازلي يوافق قوى المتحول في كثير الحدود بدءاً من القوة (n) وحتى القوة (0) بما في ذلك المعاملات المعدومة للمتحول ، حيث (n) أعلى رتبة للمتحول في كثير الحدود وذلك للتوافق ما الأسلوب الذي يتعامل معه برنامج الماتلاب في حل كثيرات الحدود.

مثال بسيط :

f(t) = $7t^4+6t^3+3t^2+t^1+5$

يتم تمثيل كثير الحدود هذا بالمصفوفة :

$$A = \begin{bmatrix}7 & 6 & 3 & 1 & 5\end{bmatrix}$$

٢,٤ العمليات الحسابية على كثير الحدود

١,٢,٤ إيجاد جذور كثير حدود

إن التابع roots هو التابع الخاص في الماتلاب الذي يمكن عن طريقه إيجاد جذور كثير حدود ، حيث يعطي هذا التابع جذور كثير الحدود على شكل شعاع عمود وعناصر هذا الشعاع هي جذور كثير الحدود.

▪ مثال (١) :

```
%f(t) = 7*t^4+6*t^3+3*t^2+t^1+5;
A = [7 6 3 1 5];
A_roots = roots(A)
```

$$\begin{matrix} -0.8301 + 0.6636i \\ -0.8301 - 0.6636i \\ 0.4015 + 0.6864i \\ 0.4015 - 0.6864i \end{matrix}$$

▪ مثال (٢) :

```
%f(t) = 7t^3+5t^2-3t^1+10
A = [7 5 -3 10];
A_roots = roots(A)
```

$$\begin{matrix} -1.5683 \\ 0.4270 + 0.8535i \\ 0.4270 - 0.8535i \end{matrix}$$

٢,٢,٤ إيجاد كثير حدود انطلاقاً من جذوره

باستخدام التابع الخاص poly يمكن إيجاد كثير حدود إنطلاقاً من جذوره ، أي أن وظيفة هذا التابع معاكسة تماماً للتابع سابق الذكر.

▪ مثال (١) :

```
A = [1 3 4 5 6];
roots(A)
```

$$\begin{matrix} -1.6665 + 0.7104i \\ -1.6665 - 0.7104i \\ 0.1665 + 1.3418i \\ 0.1665 - 1.3418i \end{matrix}$$

```
poly(ans)
```

$$ans = [1 \quad 3 \quad 4 \quad 5 \quad 6]$$

▪ مثال (2) :

```
A = [3 5 6 7 8 9];
roots(A)
```

$$\begin{array}{c} 0.5275 + 1.0679i \\ 0.5275 - 1.0679i \\ -1.3197 \\ -0.7009 + 1.0541i \\ -0.7009 - 1.0541i \end{array}$$

```
poly(ans)
```

$$ans = [1 \quad 1.6667 \quad 2 \quad 2.333 \quad 2.6667 \quad 3]$$

نلاحظ من المثال الثاني أن هناك كثير حدود غير كثير الحدود الذي ادخلناه أولاً له نفس الجذور ، وبالتالي ليس بالضرورة الحصول على التابع المدخل مسبقاً عند استدعاء (إعادة توليد) كثير الحدود الجديد من التابع (poly) ، كمال هو جدير بالذكر أنه هذا تابع فيه نسبة خطأ لا تتجاوز (1/1000000) قد تظهر في بعض الحالات.

٣,٢,٤ حساب قيمة كثير حدود عند قيمة معينة

تتم هذه العملية في الماتلاب باستخدام التابع الخاص polyval(p,x) حيث يعبر p النسق (المصفوفة) الممثل لكثير الحدود ، والرمز x يعبر عن قيمة المتحول المراد حساب كثير الحدود عنده.

▪ مثال (١) :

G(x) = x^2+x^1+1

```
p = [1 1 1];
x = 3;
gx = polyval(p,x)
```

gx= 13

كما يمكن بطريقة ثانية إيجاد قيمة كثير الحدود لأجل قيمة معينة باستخدام التابع الخاص subs وبالاستعانة بالتعليمة syms حيث يقوم هذا الأخير بتحويل قيمة المتحول x إلى رمز يمكن التعامل

معه من قبل البرنامج والتعرف عليه ، والأمر subs الذي يقوم بحساب القيمة لكثير الحدود عند قيمة المتحول المطلوبة (أي أن التعليمة syms وظيفتها تعريف المتغير x)

- مثال (2) :

```
syms x
gx = x^2 + x + 1;
subs(gx,3)
```

ans = 13

- مثال (3) :

```
syms x y
gx = y*x^2 + x*y + 1;
subs(gx,x,3)
```

ans = 12*y + 1

```
subs(gx,y,3)
```

ans = 3*x^2 + 3*x + 1

- مثال (4) :

المطلوب حساب قيم كثير الحدود من أجل عناصر المصفوفة x

```
p=[1 1 1]
x = [2 4 6]
gx = polyval(p,x)
```

$$gx = [7 \quad 21 \quad 43]$$

٤,٢,٣ اشتقاق كثيرات الحدود

يمكن إيجاد مشتق كثير حدود باستخدام التابع polyder وتكون النتيجة عبارة عن كثير حدود يمثل مشتق كثير الحدود الأصل.

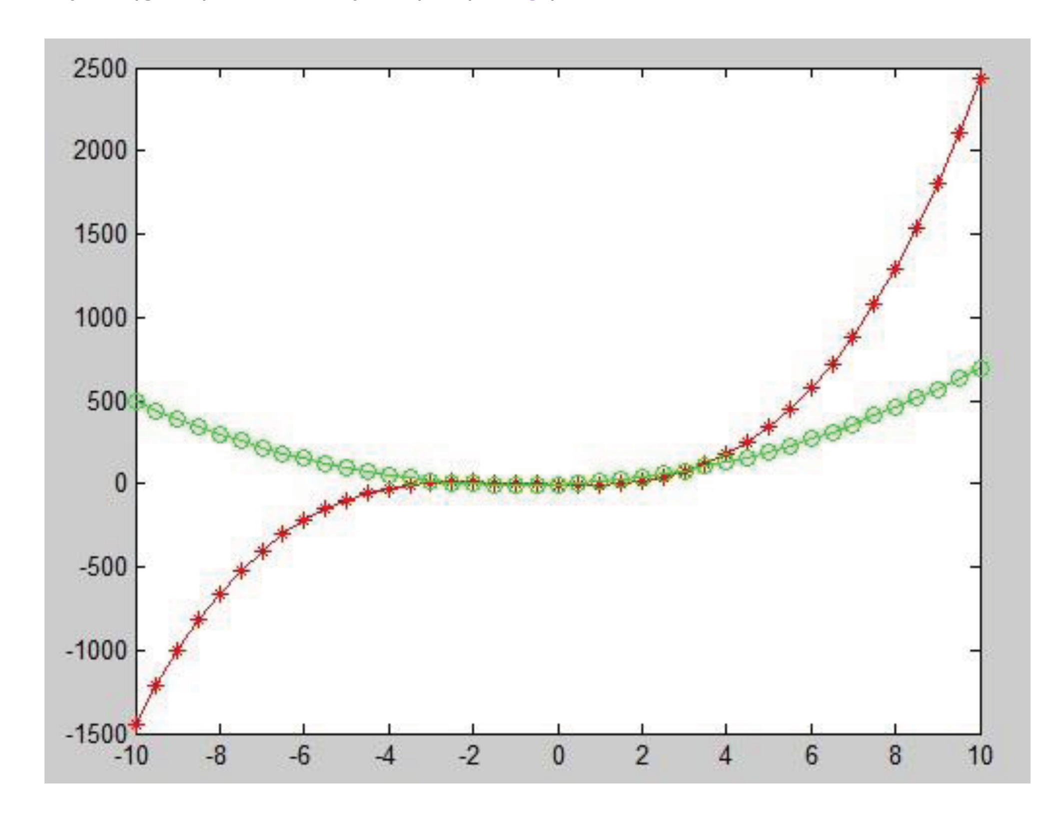
- مثال (١) :

```
gx = [2 5 -6 -5];
a = polyder(gx)
```

$$a = [6 \quad 10 \quad -6]$$

لمقارنة التابع مع مشتقه بيانياً ، يمكن رسمهما معاً على نفس المحاور (محاور الاحداثيات) بالتعليمات التالية :

```
x=-10:0.5:10;
gx=[2 5 -6 -5];
a = polyder(gx)
plot(x,polyval(gx,x),'-*r',x,polyval(a,x),'-og')
```



كما يمكن استخدام مشتق كثير الحدود بطريقة أخرى وذلك باستخدام الأمر syms والأمر
... diff(function name)

- مثال (٢) :

```
syms x
gx = 2*x^3 + 5*x^2 - 6*x - 5;
diff(gx)
```

ans = 6*x^2 + 10*x -6

- مثال (3) :

```
syms x
gx = 2*sin(2*x)*exp(x)
```

```
diff(gx)
```
ans = 4*cos(2*x)*exp(x) + 2*sin(2*x)*exp(x)

- مثال (4) :

```
syms x y
gx = 2*sin(x*y)
diff(gx,x)
```
ans = 2*cos(x*y)*y

في هذا المثال نلاحظ أن التابع diff يمكنه أيضاً حساب المشتق بالنسبة لأحد المتحولات (في حال وجود أكثر من متحول في التابع ، في مثالنا هذا قمنا بإيجاد مشتق التابع بالنسبة للمتحول (x).
كما يمكن أيضاً من خلال هذا التابع إيجاد المشتقات من المرتبة الثانية والثالثة ومافوق وذلك كما يلي

```
diff(gx,x,2)
diff(gx,x,3)
```

أي إيجاد المشتق (الثاني أو الثالث) بالنسبة للمتحول x....

٥,٢,٤ تكامل كثيرات الحدود

من المعروف أن الاشتقاق هو عملية معاكسة للاشتقاق ، ويمكننا التابع polyint الموجود في المكتبة الرياضية في الماتلاب من إيجاد تكامل تابع من خلال إدخال معاملات كثير الحدود بشكل نسق(مصفوفة) كما سبق في الأمثلة السابقة .

- مثال :

المطوب إيجاد تكامل كثير الحدود $(6x^2+10x-6)$ من أجل ثابت تكامل k = -5

```
p =[6 10 -6];
k = -5;
gx = polyint(p,k)
```
gx = $[2 \quad 5 \quad -6 \quad -5]$

كما يمكن إيجاد تكامل كثير الحدود باستخدام syms والتابع int وذلك بشكل يماثل التابع diff لإيجاد المشتق ....

```
int(gx)
```

٦,٢,٤ إيجاد معادلة كثير الحدود الملائم

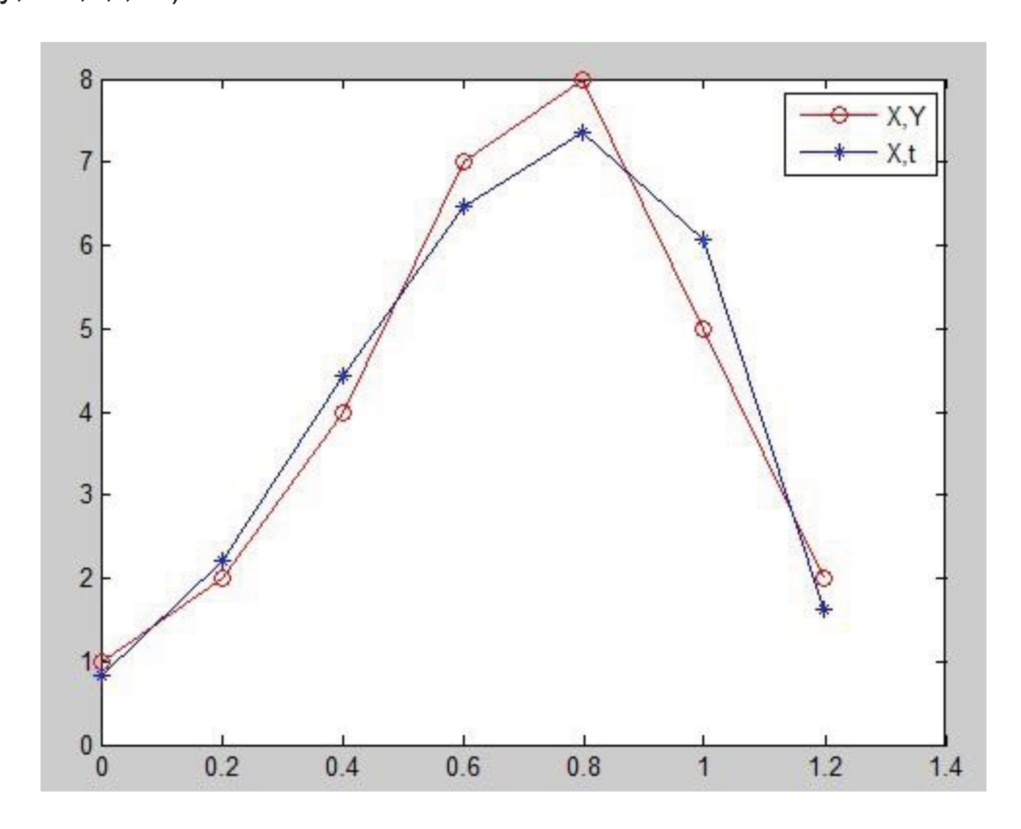
إذا كان لديك مجموعة نقاط من منحني لا تعرف معادلته وتريد ان تشكل معادلة لهذا المنحني ، فيمكن الاستعانة بالتابع polyfit للقيام بهذه العملية ، حيث أنه يعطيك معادلة كثير الحدود الملائم للمنحني بحيث أنك تختار درجة كثير الحدود الذي تريد الحصول عليه.

- مثال :

```
x=[0:0.2:1.2];
y=[1 2 4 7 8 5 2];
u=polyfit(x,y,3)
```

$$u = [-20.833 \quad 22.9167 \quad 3.1548 \quad 0.8333]$$

```
t=polyval(u,x)
plot(x,y,'-or',x,t,'-*')
```



٧,٢,٤ العمليات الرياضية على كثيرات الحدود

يمكن جمع وطرح كثيرات الحدود من خلال التعامل مع المصفوفات الممثلة لها مع ملاحظة ان تكون كلا المصفوفتين من نفس الدرجة والمثال التالي يوضح ذلك ....

▪ مثال :

$A = 2x^4 + x^3 + 3x^2 + x + 1$
$B = 4x^2 - x - 1$

الحل :

```
A = [2 1 3 1 1];
B = [4 -1 -1];
C = A + [0,0,B]
D = A - [0,0,B]
```

$$C = [2 \quad 1 \quad 7 \quad 0 \quad 0]$$
$$D = [2 \quad 1 \quad -1 \quad 2 \quad 2]$$

اما عملية الجداء فيمكن القيام بها بالاستعانة بالتابع conv مختصر كلمة (Convolution) ، حيث باستخدام هذا التابع يتم الحصول على جداء كثيرات الحدود بشرط أن تعرف كثيرات الحدود بأنساق موافقة ومعبرة عن معاملاتها.

```
conv(A,B)      %conv(B,A)
```

$$ans = [8 \quad 2 \quad 9 \quad 0 \quad 0 \quad -2 \quad -1]$$

٨,٢,٤ عمليات اخرى

- إيجاد المشتقات الجزئية(اليعقوبي)لثلاث توابع بثلاث مجاهيل [x,y,z] استعن بالتابع jacobian.

```
syms x y z
f = [x*y*z; y; x + z];
v = [x, y, z];
R = jacobian(f, v)
b = jacobian(x + z, v)
R =
[ y*z, x*z, x*y]
[ 0, 1, 0]
[ 1, 0, 1]
b =
[ 1, 0, 1]
```

- فك الأقواس و تجميع المعاملات من نفس الأس , استعن بالتابع collect.

```
syms x y
```

```
R1 = collect((exp(x)+x)*(x+2))
R2 = collect((x+y)*(x^2+y^2+1), y)
R3 = collect([(x+1)*(y+1),x+y])
return
R1 =
x^2 + (exp(x) + 2)*x + 2*exp(x)
R2 =
y^3 + x*y^2 + (x^2 + 1)*y + x*(x^2 + 1)
R3 =
[ y + x*(y + 1) + 1, x + y]
```

- فك الأقواس و نشر كثير الحدود أو المعادلة , استعن بالتابع expand .

```
syms x
expand((x-2)*(x-4))
```

الناتج:

```
ans =
x^2 - 6*x + 8
syms a b c
expand(log((a*b/c)^2))
```

الناتج:

```
ans =
log((a^2*b^2)/c^2)
```

- تبسيط التوابع و المعادلات و كثيرات الحدود , استعن بالتاب simplify .

```
1) syms a b c
simplify(exp(c*log(sqrt(a+b))))
ans =
(a + b)^(c/2)

2) syms x
S = [(x^2 + 5*x + 6)/(x + 2), sqrt(16)];
R = simplify(S)
R =
[ x + 3, 4]
```

الجدول التالي يبين أهم التوابع المستخدمة وحسب الترتيب الأبجدي

conv	ضرب كثير الحدود
deconv	قسمة كثير الحدود
poly	اكتشاف معادلة كثير حدود جذورها معلومة
polyder	مشتق كثير حدود
polyfit	إيجاد كثير الحدود الملائم
polyint	تكامل كثير الحدود
polyval	حساب قيمة كثير الحدود عند قيمة معينة
roots	إيجاد جذور كثير الحدود

ملاحظة هامة :

تمكننا المصفوفات في الماتلاب من حل جملة n معادلة بـn متغير بسهولة تامة وذلك كما يوضح المثال التالي ....

$$5x_1 - 2x_2 + x_3 = 1$$
$$x_2 + x_3 = 0$$
$$x_1 + 6x_2 - 3x_3 = 4$$

الحل :

```
clear
A = [5 -2 1;0 1 1; 1 6 -1];
B = [1;0;4];
X=inv(A)*B
```

$$X = \begin{matrix} 0.5 \\ 0.5 \\ -0.5 \end{matrix}$$

لنأخذ مثال آخر ...

$$x_1 - 5x_2 - 8x_3 + x_4 = 3$$
$$3x_1 + x_2 - 3x_3 - 5x_4 = 1$$
$$x_1 - 7x_3 + 2x_4 = -5$$
$$11x_2 + 20x_3 - 9x_4 = 2$$

الحل :

```
clear
A = [1 -5 -8 1;3 1 -3 -5;1 0 -7 2;0 11 20 -9];
B = [3;1;-5;2];
det(A)
```

X=inv(A)*B

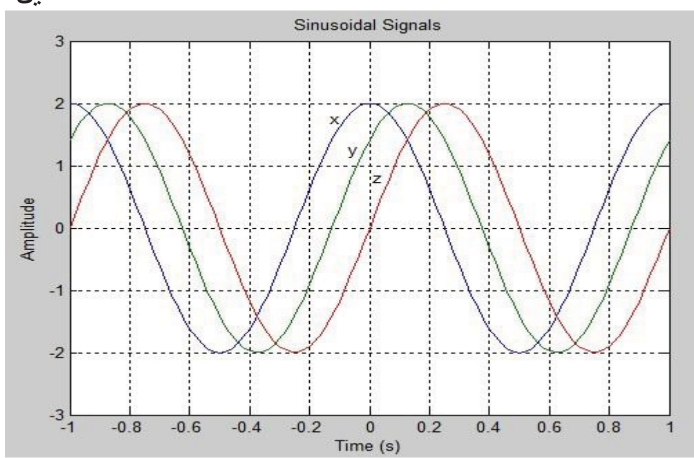
بملاحظة أن محدد المصفوفة A يساوي الصفر فهذا يعني أننا لا يمكن أن نجد مقلوب المصفوفة وبالتالي تكون جملة المعادلات السابقة غير متوافقة وليس لها حل.

تطبيقات هندسية و أمثلة عملية

١,٥ البرنامج الأول

برنامج يبين فرق الطور بين الإشارات الجيبية

```
t=linspace(-1,1,101);مصفوفة متزايدة خطياً
x=2*cos(2*pi*t);
y=2*cos(2*pi*(t-0.125));
z=2*sin(2*pi*t);
plot(t,x,t,y,t,z)
axis([-1,1,-3,3])
title('Sinusoidal Signals')
ylabel('Amplitude')
xlabel('Time (s)')
text(-0.13,1.75,'x') تسمية كل شكل
text(-0.07,1.25,'y')
text(0.01,0.8,'z')
grid on          تفعيل الشبكة
```

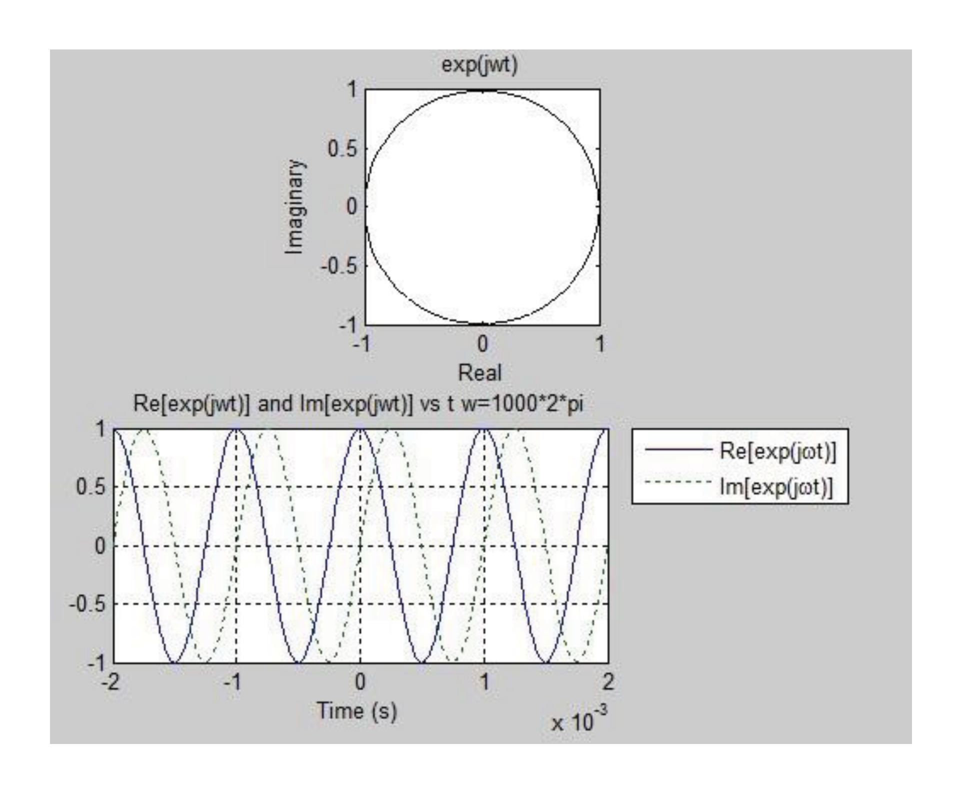


٢,٥ البرنامج الثاني

برنامج التمثيل الطوري للإشارة الجيبية أي رسم القسم الحقيقي بالنسبة للعقدي ورسم كل منهما بالنسبة للزمن ...

من المعلوم أن الإشارة الجيبية لها تمثيل عقدي يبين بالشكل التالي : $e^{j\theta}$= cos(θ) + j.sin(θ)

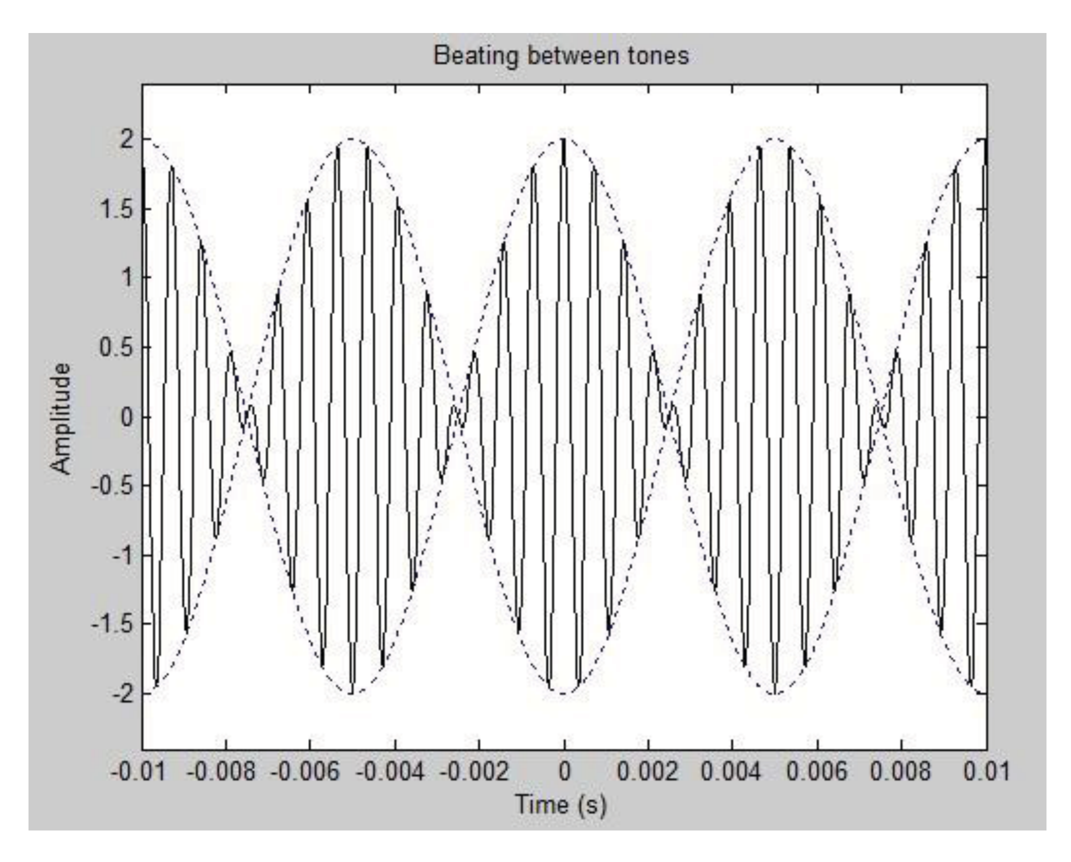
```
t=(-2e-03:0.02e-03:2e-03);
x=exp(j*2000*pi*t);    expالتابع الأسي في الماتلاب
y=real(x);  القسم الحقيقي للإشارة
z=imag(x);  (التخيلي) القسم الوهمي للإشارة
subplot(2,1,1)    تقسيم الرسم
plot(x,'-.k')
axis square     جعل المحاور مربعة الشكل
title('exp(jwt)')
xlabel('Real')
ylabel('Imaginary')
subplot(2,1,2)
plot(t,y,'-',t,z,':')
title('Re[exp(jwt)] and Im[exp(jwt)] vs t w=1000*2*pi')
xlabel('Time (s)')
grid on
legend('Re[exp(j\omegat)]','Im[exp(j\omegat)]',-1)
```



٣,٥ البرنامج الثالث

برنامج دمج نغمتين ...

```
t=linspace(-1e-2,1e-2,1001);
x=cos(2*pi*1500*t) + cos(2*pi*1300*t);نفس المطال،غير تردد
m=2*cos(2*pi*100*t);      الإشارة المرجعية (إشارة الغلاف)
plot(t,m,'b:',t,-m,'b:',t,x,'k')
axis([-0.01 0.01 -2.4 2.4])
title('Beating between tones')
xlabel('Time (s)')
ylabel('Amplitude')
```



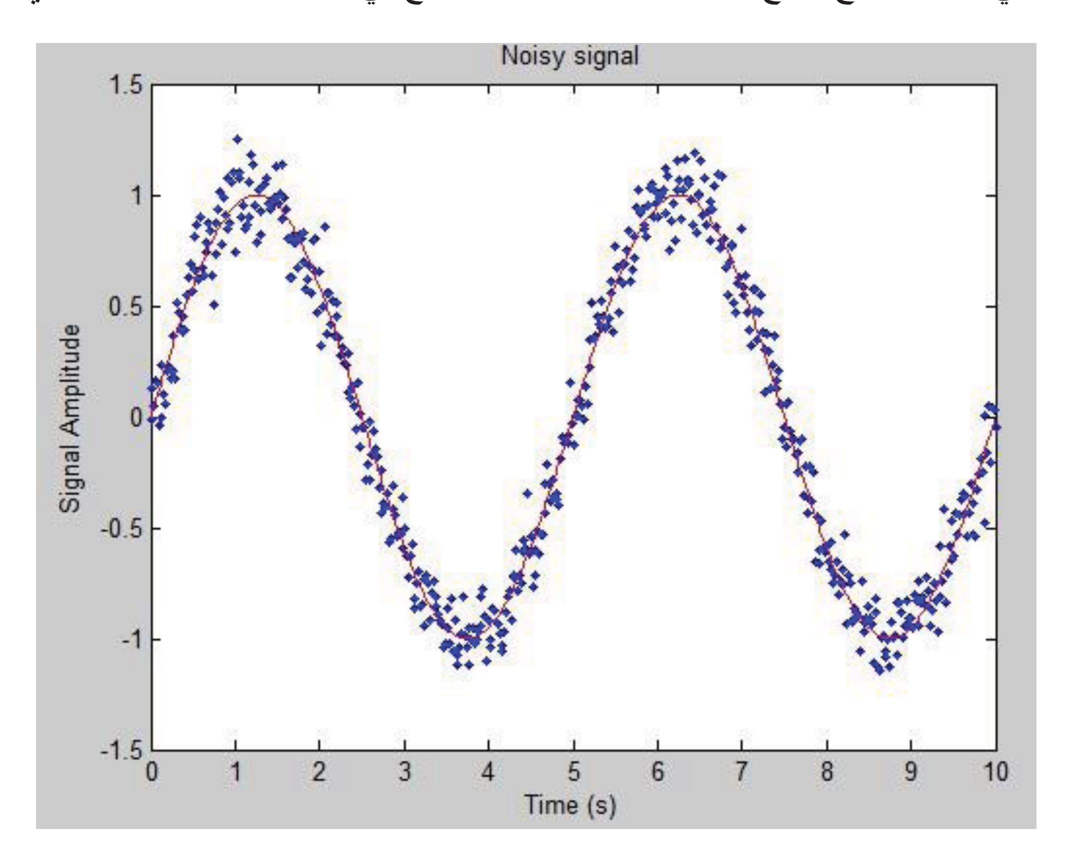
٤,٥ البرنامج الرابع

برنامج يمثل دخول الضجيج على الإشارة الجيبية ...

```
t=linspace(0,10,512)
s=sin(2*pi/5*t);     الإشارة الأصلية
n=0.1*randn(size(t));  إشارة الضجيج
x = s + n;          الإشارة المشوهة
disp('Signal to Noise Ratio(SNR), dB')
SNR = 20*log10(std(s)/std(n))   نسبة الإشارة إلى الضجيج
plot(t,x,'.',t,s,'r')
```

```
xlabel('Time (s)')
ylabel('Signal Amplitude')
title('Noisy signal')
```

استخدمنا في هذا البرنامج التابع std وتكمن وظيفة هذا التابع في حساب الانحراف المعياري.

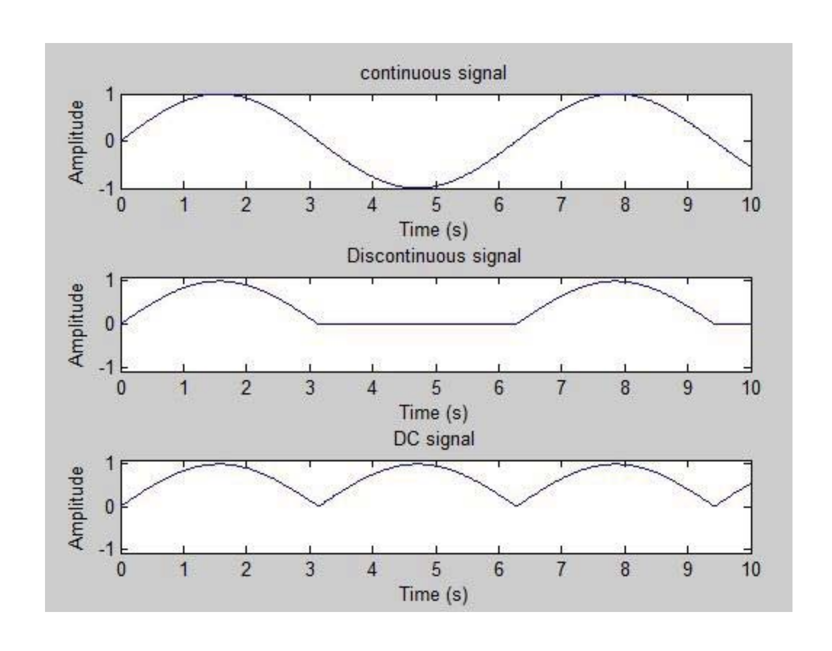


٥,٥ البرنامج الخامس

برنامج لرسم الإشارة المقومة

```
t=linspace(0,10,512)
x=sin(t);
x1=x.*(x>0);          الإشارة المقومة الأولى
x2=abs(x);            الإشارة المقومة الثانية
subplot(3,1,1)        تجزئة الرسم
plot(t,x)
xlabel('Time (s)')
ylabel('Amplitude')
title('continuous signal')
subplot(3,1,2)
plot(t,x1)
axis([0 10 -1.1 1.1])
```

```
title('Discontinuous signal')
xlabel('Time (s)')
ylabel('Amplitude')
subplot(3,1,3)
plot(t,x2)
axis([0 10 -1.1 1.1])
title('DC signal'
xlabel('Time (s)')
ylabel('Amplitude')
```



٦,٥ البرنامج السادس

برنامج لحساب المقاومة المكافئة لدارة مؤلفة من n ممانعة من الشكل$(R \pm jX)$ بحيث يحدد المستخدم عدد الممانعات الكلي في الدارة وقيمة كل ممانعة بقسميها الحقيقي والوهمي ، كما يحدد المستخدم طريقة توصيل الممانعات بشكل تسلسلي أو تفرعي وبالتالي يكون المستخدم قد حدد شكل الدارة بالكامل...

يجب في هذا البرنامج ملاحظة أن الممانعة رقم n لا يهمني طريقة توصيلها ومايهمني هو طريقة توصيل الممانعة (n-1) معها ولذلك سنلاحظ أن البرنامج لن يسأل عن طريقة توصيل الممانعة n وإنما سيبدأ السؤال بالممانعة (n-1)

```
n = 8;              فرضاً
P = ones(1,n);  مصفوفة سطرية واحدية
for i=n:-1:1
     P(1,i) = input (['Z' num2str(i) ':']); إدخال قيم الممانعات بالترتيب
end

Q = ones(1,n-1);
```

```
for j=1:n
   if j==1
      connect = input(['   Z' num2str(n-j) '   series or Parallel : ']);
       switch connect          تحديد نوع توصيل الممانعات
           case {'series' 's'}
               Q(1,n-1) = P(1,n) + P(1,n-j);
           case {'Parallel' 'P'}
               Q(1,n-1) = (P(1,n)*P(1,n-j))/(P(1,n)+P(1,n-j));
       end
   else

       if n==j
           break;                         إيقاف البرنامج عندما تصل الحلقة لقيمة مساوية لعدد الممانعات في الدارة
       else
          connect = input(['   Z' num2str(n-j) '   series or Parallel : ']);
            switch connect
               case {'series' 's'}
                  Q(1,n-j) = Q(1,n-j+1) + P(1,n-j);
               case {'Parallel' 'P'}
                  Q(1,n-j) = Q(1,n-j+1)*P(1,n-j)/(Q(1,n-j+1)+P(1,n-j));
            end
       end
   end
end
Zeq = Q(1,1);هذا العنصر من المصفوفة يمثل الممانعة الكلية المكافئة للدارة
```

تنفيذ البرنامج :

```
MATLAB 7.6.0 (R2008a)
File  Edit  Debug  Parallel  Desktop  Window  Help
Current Direc
Shortcuts  How to Add  What's New
New to MATLAB? Watch this Video, see Demos, or read Gettin
Z8:10+5j
Z7:12
Z6:10+7j-12j
Z5:3+j
Z4:4
Z3:10j
Z2:-15j
Z1:10
   Z7   series or Parallel : 'Parallel'
   Z6   series or Parallel : 'series'
   Z5   series or Parallel : 'P'
   Z4   series or Parallel : 'P'
   Z3   series or Parallel : 's'
   Z2   series or Parallel : 'P'
   Z1   series or Parallel : 's'

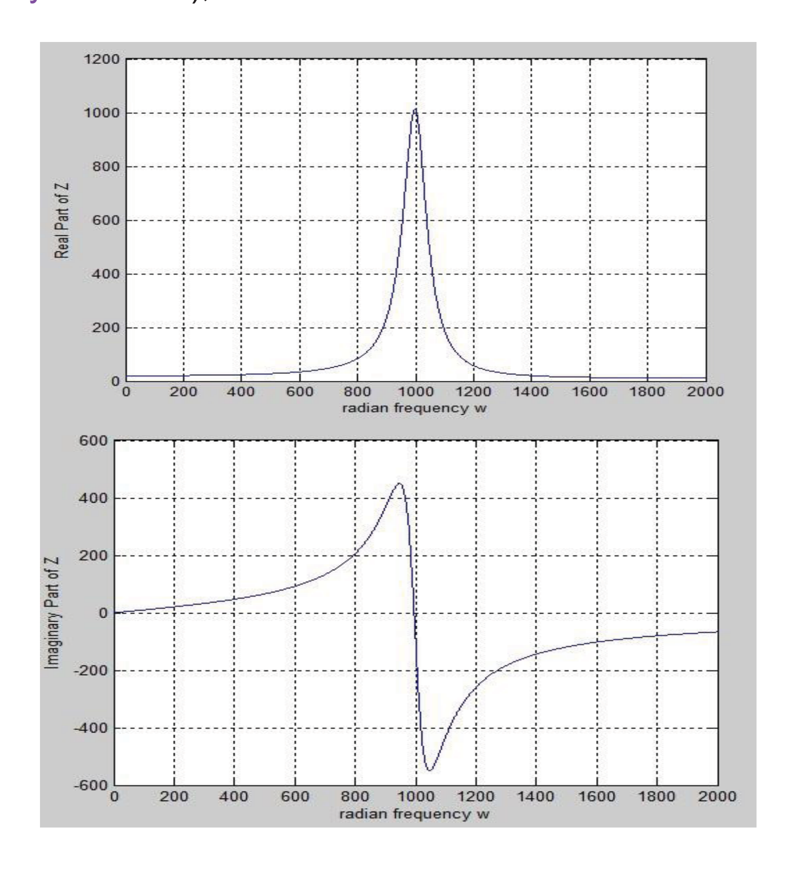
Zeq =

  24.2528 +27.3446i

>> |
```

٧,٥ البرنامج السابع

```
w=0:1:2000;
z=(10+(10.^4-j.*(10.^6./(w)))./(10+j.*(0.1.*w-10.^5./w)));
Real=real(z);
plot(w,Real);
xlabel('radian frequency w');
ylabel('Real Part of Z');
grid on
pause(5)
imagine=imag(z);
plot(w,imagine);
grid on
xlabel('radian frequency w');
ylabel('Imaginary Part of Z');
```

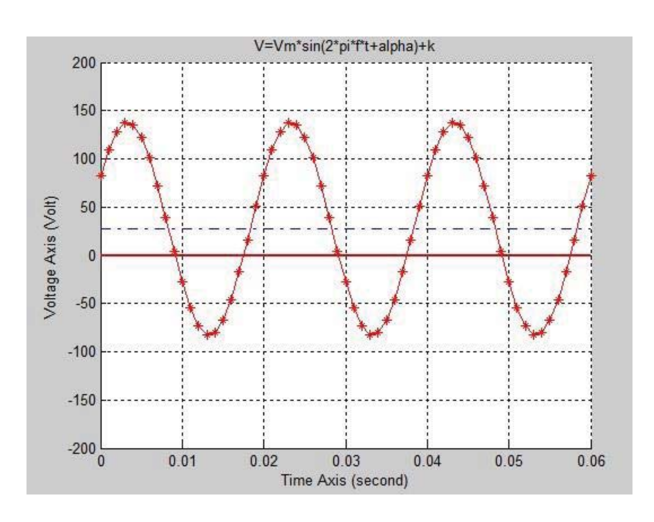


٨,٥ البرنامج الثامن

برنامج لرسم تغيرات إشارة التوتر المتناوب V(t)=Asin(wt+alpha)+k مع الزمن.

```
f=50
Vm=110;
```

```
alpha=pi/6;
k=0.25*Vm;
T=1/f;
step=T/20;
t1=0
t2=3*T;
t=t1:step:t2;
V=Vm*sin(2*pi*f*t+alpha)+k;
fig=figure;
ax=axes;
set(ax,'xlim',[t1 t2],'ylim',[-200 200]);
title('V=Vm*sin(2*pi*f*t+alpha)+k');
xlabel('Time Axis (second)');
ylabel('Voltage Axis (Volt)');
hold on;
grid on;
ta=line([t1 t2],[0 0]);
set(ta,'color','r','linewidth',2);
Vo=line([t1 t2],[k k]);
set(Vo,'color','b','linestyle','-.');
plot(t,V,'r-*');
```



الجدول التالى يبين أهم الأوامر والتوابع فى الماتلاب

Abs	القيمة المطلقة أو طويلة العدد العقدي
Angle	زاوية الطور
Ans	الجواب عند حدم إسناد التعبير
Atan	Arc tg

Axis	تدريج المحور يدوياً
Bode	رسم مخطط بود
Clc	مسح فضاء العمل
Conj	مرافق العدد العقدي
Cos	تجيب الزاوية
Cosh	التجيب القطعي
Det	إيجاد معين مصفوفة
Diag	إيجاد قطر مصفوفة
exit	إنهاء البرنامج
exp	التابع الأسي
expm	رفع مصفوفة لقوة
eye	المصفوفة الحيادية
Format long	إعطاء ١٥ مرتبة بعد الفاصلة العشرية
Format long e	إعطاء ١٥ مرتبة إضافية للقوة
Format short	إعطاء ٥ مراتب
Format short e	إعطاء ٥ مراتب إضافية للقوة
freqs	التحول إلى المستوي اللابلاسي
freqz	التحول إلى المستوي z
grid	رسم خطوط الشبكة
hold	تثبيت المخطط المرسوم على الشاشة
Real	الجزء الحقيقي لعدد عقدي
imag	الجزء التخيلي لعدد عقدي
inf	عدد لا نهائي
inv	مقلوب مصفوفة
length	طول شعاع
linspace	فضاء شعاعي خطي
log	اللوغاريتم الطبيعي (النبري)
Loglog	الرسم على محورين لوغاريتميين
Log m	أخذ لوغاريتم عناصر مصفوفة
Logspace	فضاء شعاعي لوغاريتمي
Log 10	اللوغاريتم العشري
Max	القيمة العظمى
Mean	القيمة المتوسطة
Medium	القيمة المتوسطة التربيعية
Min	القيمة الصغرى

Nan	ليس رقم (يعامل لمحرف)
Nyquist	رسم مخطط نايكويست
Pi	Pi = 3.14
Plot	الرسم في المستوي الديكارتي
Polar	الرسم في المستوي القطبي
Prod	حاصل ضرب العناصر
Rand	توليد أعداد عشوائية أو مصفوفة
Rank	حساب رتبة مصفوفة
Rem	باقي قسمة
Residue	توسيع الجزء العشري
Rlocus	رسم الجذور الصفرية
Semilogx	رسم نصف لوغاريتمي \ المحور x هو اللوغاريتمي
Semilogy	رسم نصف لوغاريتمي \ المحور y هو اللوغاريتمي
Sign	إسناد عدد
Sqrtm	مصفوفة الجذور التربيعية
Std	الانحراف المعياري
Step	رسم منحني الاستجابة الواحدية
Who	إدراج المتحولات الموجودة في الذاكرة

الجدول التالى يبين أهم العمليات والعلاقات فى الماتلاب

الجمع	+
الطرح	-
الضرب	*
الرفع لقوة	^
منقول مصفوفة	'
أصغر من	<
أصغر أو يساوي	<=
أكبر من	>
يساوي	==
لايساوي	~=
AND	&
OR	!
NOT	~

الجدول التالى يبين أهم الرموز الخاصة فى الماتلاب

[ ]	يستخدم لتشكيل المصفوفات
( )	التعبير الرياضي عن الأشعة
'	الفصل بين التعليقات وبين مناقشة الوظائف
;	سطروإنهاء السطر ووقف الكتابة ومنع الأمر المنفذ قبلها من الظهور
:	إقرار امتداد شعاع
%	إشارة تدل على أن ما بعدها تعليق يهمله البرنامج

٩,٥ البرنامج التاسع

اكتب برنامج لحل جملة معادلات مكونة من (m) مجهول بحيث يقوم المستخدم بإدخال قيم ثوابت المعادلات بالترتيب فالتالي:

$$A_{11} \rightarrow A_{1m}$$

$$A_{21} \rightarrow A_{2m}$$

$$A_{m1} \rightarrow A_{mm}$$

وبعد ذلك يقوم المستخدم بإدخال قيم الثوابت.

ومن ثم و بعد إيجاد حل جملة المعادلات m ، يقوم المستخدم بالقيام بالعمليات التالية على مصفوفة الثوابت (أمثال المجاهيل) وذلك بعد اختيار العملية من قائمة منسدلة :

- إيجاد أصغر قيمة وأكبر قيمة لعناصر المصفوفة.
- إيجاد المتوسط الحسابي لمجموع عناصر المصفوفة ، ومن ثم ضرب عناصر المصفوفة بالمتوسط الحسابي الناتج .
- إضافة عمود جديد قيمه تساوي جداء قيم عناصر العمود الأول من المصفوفة بالعمود الأخير منها.
- قلب المصفوفة(يمين يسار)و قلب المصفوفة(اعلى أسفل.)
- جعل كافة العناصر فوق القطر الرئيسي أصفار.
- إيجاد مجموع عناصر القطر الرئيسي(بتابع واحد فقط).

الحل:

```
clc
clear
%----------------------- Initialization Part ---------------------------
% get the number of equations from the user
```

```
disp('This script for solve m-equation with m variable'); % Ax = B
m = input('m = ');
while isempty(m) || ~isnumeric(m)
m = input('m = ');
end
% Check m value, it must be positive
while m <= 0
disp('m must be positve ...');
m = input('m = ');
end
% define an array of variables coefficients
A = ones(m,m);
% define an array of constants
B = ones(1,m);
% define an inverse array of variables coefficients array
inv_A = ones(m,m);
% get variables coefficients from the user
current = 0;
for i = 1:m
for j = 1:m
A(i,j)=input(['A(' num2str(i) ',' num2str(j) '):']);
% while (~isnumeric(A(i,j)) || isempty(A(i,j)) )
% A(i,j) = input('');
% end
end
end
% get constants from the user
for i = 1:m
B(1,i)=input(['B(' num2str(i) '):']);
end
%---------------------- Calculation Part -------------------------
% Calc variables coefficients array determinant
Det_A = det(A);
% Calc the inverse variables coefficients array
inv_A = inv(A);
% find the solution and display it
x = inv_A*B';
```

```
%----------------------- Display Results Part ---------------------------
% display the variables coefficients array determinant
disp('Matrix A determinant is :')
display(A);

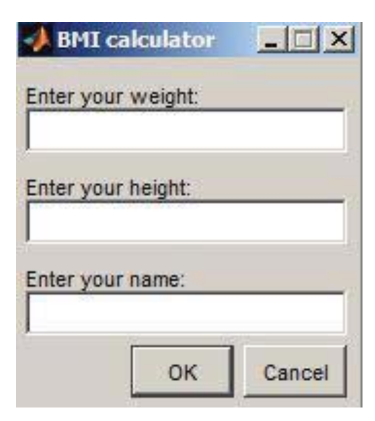
% display the constants array
display(B);
% display the inverse of variables coefficients array
disp('The inverse of A is :');
display(inv_A);
disp('The solution is :');
display(x);

%******************************** Menu Part *****************************
t = menu('What do you want to do with (A) Matrix:','Max Value','Min
Value','Avarage', 'Flip (up down)', 'Flip (Left right)', right)','Make upper
triangular part of the matrix','Sum the diag','Quit');
if t==1
Max = max(A)
elseif t==2
Min = min(A)
elseif t==3
Mean = mean(A)
elseif t==4
Flip= flipud(A)
elseif t==5
Flip= fliplr(A)
elseif t==6
Upper_Trianguler= tril(A)
elseif t==7
trace(A);
end
```

١٠,٥ البرنامج العاشر

اكتب برنامج يحدد يدخل به المستخدم درجة الحرارة بالسيلسيوس ليحولها له البرنامج للفهرهايت ويظهرها على الشاشة.

يتم إدخال قيمة درجة الحرارة من خلال نافذة كالمبينة بالشكل:



ويتم إظهار النتيجة في Command windows.

طور هذا البرنامج بين يظهر درجات الحرارة بالفهرهايت المقابلة لدرجات الحرارة بالسيلسيوس في المجال $20 < T < 30$ وبالشكل التالي:

```
Celsius      Fahrenheit
20.00        68.00
21.00        69.80
...
30.00        86.00
```

الحل :

```
clear
clc
prompt={'Enter Temperature :'};
% Create all your text fields with the questions specified by the
variable prompt.
title='Temp. Converter';
% The main title of your input dialog interface.
answer=inputdlg(prompt,title);
a = str2num(answer{1});
b=a*9/5+32;
disp(['T= ' num2str(b) ' F'])
disp('Celsius      Fahrenheit')
```

```
for i=0:10
c=(20+i)*9/5+32;
disp([num2str(20+i) ' c' '          ' num2str(c) ' F'])
end
```

١١,٥ البرنامج الحادي عشر

المطلوب كتابة برنامج يتيح للأستاذ إدخال أسماء الطلاب وعلاماتهم (في امتحان ما) ومن ثم إظهار اسم الطالب الذي حصل على اعلى علامة واسم الطالب الذي حصل على أقل علامة وكذلك المتوسط الحسابي للعلامات.

الحل:

```
clc
clear
n = input('Enter the number of students : ');
while ((isempty(n)) || (n <= 0))
n = input('Enter the number of students : ');
end
Students = cell(n,2);
Marks = 0;
for i = 1:n
Students{i,1} = input(['Enter the name of Student(' num2str(i) ') : '],'s');
Students{i,2} = input(['Enter the score of Student(' num2str(i) ') : ']);
Marks = Marks + Students{i,2};
end
Students
sortedCellArray = sortrows(Students,-2);
display(['The first student is :', sortedCellArray(1,1)]);
display(['The last student is :', sortedCellArray(n,1)]);
Average = Marks/n;
sprintf('Average of marks is %f \n',Average)
```

الحل بطريقة ثانية:

```
clc
clear
n=input('number of student :');
name={};
grade=ones(n,1);
```

```
for i=1:n
name(i)={input('Name : ','S')};
grade(i)=input('Grade : ');
end
max=0; w=0;
min=100; k=1;
for j=1:n
if min>grade(j)
min=grade(j);
k=j;
end
end;
disp('name of student min grade');
disp(name(k));
for m=1:n
if max<grade(m)
max=grade(m);
k=m;
end
end;
disp('name of student max grade');
disp(name(k));
disp('average is: ');
w=mean(grade);
disp(w);
```

١٢,٥ البرنامج الثاني عشر

اكتب برنامج يمكن المستخدم من حماية برامجه الشخصية بحيث يعطي للمستخدم اسم مستخدم وكلمة مرور وفي حال إدخال الاسم وكلمة المرور بشكل صحيح فإن البرنامج يفتح ويعمل بشكل صحيح وفي حال كانت خاطئة لا يدخل المستخدم ولا يفتح البرنامج وفي حال إدخال المستخدم لكلمة السر ثلاث مرات بشكل خاطئ يغلق البرنامج. (تماماً كالدخول لحساب الفيس بوك أو المسنجر)
استعن بالمصفوفات للقيام بذلك(مصفوفة محارف).

الحل :

```
Username='MATLAB';
Password='MATLAB';
user=input('Enter your user name : ','s');

pass=input('Enter your password : ','s');
for i=1:length(user)
if length(user)~=length(Username)
disp('Wrong User Name.')
break
elseif user(i)~=Username(i)
disp('Wrong User Name.')
break
end
con=1;
end
if con==1
for j=1:length(pass)
if length(pass)~=length(Password)
disp('Wrong Password.')
break
elseif pass(j)~=Password(j)
disp('Wrong Password')
break
end
if j==length(pass)
disp('Welcome');
end
end
end
```

١٣,٥ البرنامج الثالث عشر

طور برنامج الحماية الذي قمت به في الدروس الماضية مضيفاً إليها ما يلي وذلك عن إدخال اسم مستخدم وكلمة مرور صحيحتين:

اكتب برنامج يتيح للمستخدم إدخال تابع من الدرجة الثانية من الشكل($f(x)=ax^2+bx+c$) حيث يحدد المستخدم الثوابت ويقوم برسم تغيرات التابع , ومشتقه وتكامله. اجعل مجال الرسم بحيث (x=0:0.01:5).

الحل:

```
Username='MATLAB';
Password='MATLAB';
user=input('Enter your user name : ','s');
pass=input('Enter your password : ','s');
for i=1:length(user)
if length(user)~=length(Username)
disp('Wrong User Name.')
break
elseif user(i)~=Username(i)
disp('Wrong User Name.')
break
end
con=1;
end
if con==1
for j=1:length(pass)
if length(pass)~=length(Password)
disp('Wrong Password.')
break
elseif pass(j)~=Password(j)
disp('Wrong Password')
break
end
if j==length(pass)
a=input('Enter a (ax^2+bx+c) : ');
b=input('Enter b (ax^2+bx+c) : ');
c=input('Enter c (ax^2+bx+c) : ');
f0=[a b c];
f1=polyder(f0);
f2=polyint(f0);
x=0:0.01:5;
plot(x,polyval(f0,x),'-k',x,polyval(f1,x),'-- r',x,polyval(f2,x),':b');
legend('f(x)','diff','int');
title('The Plot');
xlabel('x');
ylabel('y');
```

```
grid on;
end
end
end
```

١٤,٥ البرنامج الرابع عشر

التابع المبين يمثل منحني التزايد السكاني في أمريكا بين العام 1791 والعام 2000 , والمطلوب رسم تغيرات التابع بالنسبة للزمن, حيث t تمثل التاريخ.

الحل:

$$P(t) = \frac{197273000}{1+ e^{-0.03134(t-1913.25)}}$$

```
clear
clc
syms t;
p=197273000/(1+exp(-0.03134*(t-1913.25)));
t=1790:1:2000;
plot(t,subs(p,t),'-r')
ylabel('Number of People (person)')
xlabel('Date (Year)')
```

١٥,٥ البرنامج الخامس عشر

اكتب برنامج يرسم الإشارة المتناوبة ((sin(wt+phase) ويرسم الإشارة المقومة (تقويم موجة كامل) على شكلين بنفس النافذة , بحيث يحدد المستخدم تردد العمل ومجال الزمن كذلك زاوية الطور phase وبعد ذلك يظهر الرسم (قبل وبعد التقويم ,) اجعل البرنامج بعد كل تنفيذ للبرنامج يسأل المستخدم في حال رغبته بإعادة البرنامج من جديد وفي حال الضغط المستخدم Y أي نعم يتم إعادة الطلب من المستخدم أن يدخل قيم الثوابت وعدا ذلك يغلق البرنامج.

الحل:

```
for i=1:inf
f=input('Enter Freq. :');
t1=input('Enter t start :');
t2=input('Enter t stop :');
p=input('Enter phase :');
t=linspace(t1,t2,1000);
w=2.*pi.*h;
y=sin(w.*t+p);
```

```
subplot(2,1,1)
plot(t,y,'g')
subplot(2,1,2)
plot (t,abs(y),'r')
m=input('if you want to complete enter Y , else enter N' , 's');
if m~='Y';
break
end
end
```

١٦,٥ البرنامج السادس عشر

اكتب برنامج يرسم تغيرات أحد التوابع الجيبية الثلاثة
(cos(5t), sin(5t+2pi/3), tan(7t-2pi/3)) حيث يختار المستخدم التابع فيُرسم من أجل:

- مجال للزمن ثابت(حدده بنفسك.)
- مجال يحدده المستخدم بحيث يحدد($t=t_{min}:step:t_{max}$) .

الحل:

```
t1=input('please enter t1 : ');
t2=input('please enter t2 : ');
steps=input('enter steps : ');
x=t1:steps:t2;
const_x = 1:100;
s = input('Please enter the number of the function u want :1-
sin(5t+2pi/3) , 2-cos(5t),3-tan(7t-2pi/3)')
switch s
case 1
g=sin(5*x+2*pi/3);
h=sin(5*const_x+2*pi/3);
case 2
g=cos(5*x);
h=cos(5*const_x);
case 3
g=tan(7*x-2*pi/3);
h=tan(7*const_x-2*pi/3);
otherwise
g = 0;
h = 0;
disp('please enter 1 ,2 or 3');
```

```
end
plot(x,g)
pause
plot(const_x,h)
```

١٧,٥ البرنامج السابع عشر

عرف مصفوفة (1000,4) بحيث تكون قيم الأعمدة محققة التوابع:

العمود الرابع	العمود الثالث	العمود الثاني	العمود الأول
4+10t	$4t + 5t^2$	$5t^2$	t = 0:1000

المطلوب رسم تغيرات كلاً من العمود الثاني والثالث والرابع مع الزمن(العمود الأول) بحيث تؤخذ قيمة كل ٣٠ ثانية <رسم تغيرات التوابع الثلاثة مع الزمن > :

- اجعل البرنامج يحدد كامل القيم للمصفوفة.
- اختبر شرط الزمن 01 (ثانية)ومضاعفاتها.
- ارسم تغيرات بالنسبة للمصفوفة الاولى(الزمن.)
- استعن بالتابع subplot للرسم على واجهة وحيدة.
- اجعل البرنامج يظهر:
  - أكبر قيمة للعمود الثاني.
  - أصغر قيمة للعمود الثالث.
  - المتوسط الحسابي للعمود الرابع

(لاحظ أن العمود الرابع هو مشتق الثالث)

الحل :

```
clear
clc
a=zeros(1000,4);
for i=1:1000
a(i,1)=i-1;
end
for j=1:1000
a(j,2)=5.*a(j,1).^2;
end
for k=1:1000
a(k,3)=4.*a(k,1)+a(k,2);
end
for l=1:1000
a(l,4)=4+10.*a(l,1);
end
```

```
a
b=zeros(1,4);
for m=2:1000
if rem(a(m,1),30)==0
b(m,1)=a(m,1);
b(m,2)=a(m,2);
b(m,3)=a(m,3);
b(m,4)=a(m,4);
end
end
x=nonzeros(b(:,1))';
y1=nonzeros(b(:,2))';
y2=nonzeros(b(:,3))';
y3=nonzeros(b(:,4))';
subplot(3,1,1)
plot(x,y1)
title('4t+5t^2')
xlabel('time [s]')
subplot(3,1,2)
plot(x,y2)
xlabel('time [s]')
title('5t^2')
subplot(3,1,3)
plot(x,y3)
title('4+10t')
xlabel('time [s]')
max2=0;
for o=1:1000
if a(o,2)>max2
max2=a(o,2);
end
end
min3=0;
for p=1:1000
if a(p,3)<min3
min3=a(p,3);
end
```

```
end
sum4=0;
for q=1:1000
sum4=sum4+a(q,2);
end
max2
min3
sum4
```

ملاحظة : يمكن الحصول على أكبر قيمة و أصغر قيمة و كذلك المجموع و المتوسط الحسابي الاستفادة من التوابع المتوفرة في مكتبة الماتلاب.

١٨,٥ البرنامج الثامن عشر

احسب التكامل بالطريقة التي تجدها مناسبة:

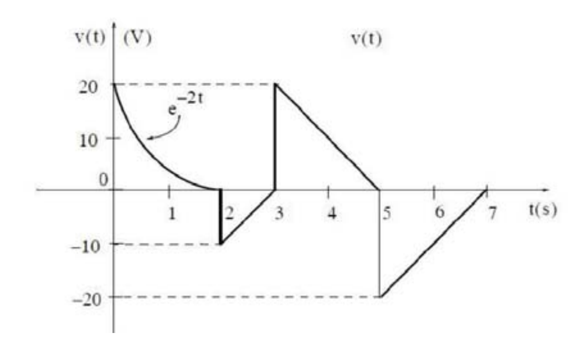
$$\int_{-\infty}^{+\infty} t^2 e^{-t}(t-2)dt$$

الحل:

```
clear
clc
syms t;
f=t^2*exp(-t)*(t-2);
int(f,-inf,inf)
```

١٩,٥ البرنامج التاسع عشر

استنتج سلسلة التوابع الرياضية المعبرة عن الرسم المبين بالرسم ومن ثم ارسمه بالطريقة التي تجدها مناسبة في الماتلاب($0 < t < 7$).



الحل:

```
t1=0:0.005:2;
y1=exp(-2*t1+3);
t2=2:3;
y2=cos(t2)*5*pi+15;
t3=3:5;
plot(t1,y1,'-k',t2,-y2,'-k')
y3=line([2 2],[-8.5 0.4],'color','k');
y4=line([3 3],[0.5 19.3],'color','k');
y5=-line([5 3],[0.5 19.3],'color','k');
y6=-line([5 5],[0.5 -20],'color','k');
y7=-line([7 5],[0.5 -20],'color','k');
grid on;
xlabel('t(s)');
ylabel('v(t)');
axis([0 7 -20 20])
```

٢٠,٥ البرنامج العشرين

المطلوب رسم تغيرات التابع $y = e^{-1.2x}sin(10x + 5)$ من أجل مجال $1 \le x \le 5$ وتغيرات التابع $y = |x^3-100|$ من أجل مجال , $-6 \le x \le 6$ وذلك بتقسيم لوحة الرسم للوحتين.

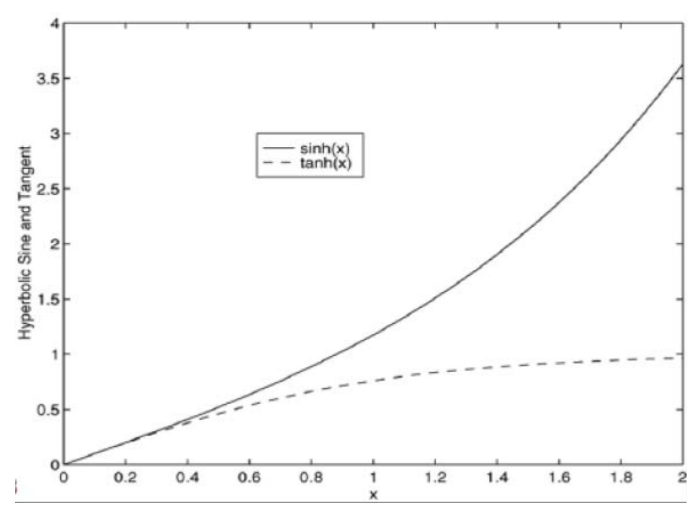
الحل:

```
y = exp(-1.2*x).*sin(10*x+5);
subplot(1,2,1)
plot(x,y)
axis([0 5 -1 1])
x = [-6:0.01:6];
y = abs(x.^3-100);
subplot(1,2,2)
plot(x,y)
axis([-6 6 0 350])
```

٢١,٥ البرنامج الحادي والعشرين

المطلوب رسم تغيرات التوابع مع تسمية الرسم كما هو موضح بالشكل التالي:

الحل:




```
X=[0:0.01:2];
y=sinh(x);
z=tanh(x);
plot(x,y,x,z,'--')
xlabel('x')
ylabel('Hyperbolic Sine and Tangent')
legend('sinh(x)','tanh(x)')
```

٢٢,٥ البرنامج الثاني والعشرين

إذا علمت أن تسارع الجاذبية الأرضية , g=9.81 ا لمطلوب رسم علاقة السرعة v=g.t مع الزمن و لمدة زمنية tf يحددها المستخدم و بخطوة للزمن مساوية tf/500.

الحل:

```
% Program falling_speed.m
% tf= final time (in seconds)
% Output Variables:
% t = array of times at which speed is
% computed (in seconds)
% v = array of speeds (meters/second)
% Parameter Value:
g = 9.81;
% Acceleration in SI units
% Input section:
tf= input('Enter final time in seconds:');
dt= tf/500;
```

```
% Create an array of 501 time values.
t=[0:dt:tf];
v = g*t;
Plot(t,v)
xlabel('t(s)')
ylabel('vm/s)')
```

٢٣,٥ البرنامج الثالث والعشرين

المطلوب إيجاد حل المعادلات التالية:

$$\sqrt{x^2 + y^2}$$

```
>> syms x y
>> s = x + y;
>> r = sqrt(x^2 + y^2);
```

$$A = x.\,^\wedge((0{:}\,3)' * (0{:}\,3))$$

```
>> n = 3;
>> syms x;
>> A = x.^((0:n)'*(0:n))
```

المطلوب إيجاد منشور تايلور للتابع الأسي e^x

```
>> syms x
>> f = exp(x);
>> taylor(f,4)
```

المطلوب إيجاد تحويل لابلاس و تحويل لابلاس العكسي للتابع e^{-bt} و التابع sin(bt) .

```
>>syms b t
>>laplace(exp(-b*t))
>>laplace(sin(b*t))
>>ilaplace(1/(s+b))
>>ilaplace(b/(s^2+b^2)
```

٢٤,٥ البرنامج الرابع والعشرين

اكتب برنامج بسيط يمثل حركة مصعد لـ ١٠ طوابق حيث يحدد المستخدم الطابق الموجود فيه و كذلك الطابق الذي يريد التوجه إليه , ليقوم البرنامج بمحاكاة المصعد بين الطوابق بمعدل 1 ثانية لكل طابق.

```
f1 = input('You are in floor number : ');
f2 = input('You want to go to floor number : ');
l = line([0.5 0.5],[-0.1 10.1]);
set(l,'linewidth',5,'color','k');
for i = f1:f2
l = line([0.5 0.5],[-0.1 10.1]);
set(l,'linewidth',5,'color','k');
e = line([0.5 0.5],[i-0.1 i+0.1]);
set(e,'linewidth',5,'color','g');
pause(1);
end
```

كيف يمكن تغيير خلفية نافذة الرسم لتصبح بلون مختلف عن اللون الأبيض, وكذلك كيف يمكن جعلها صورة ما بدلاً من لون واحد(صورة بكلا الحالتين)استعن بالتوابع imshow , imread .

الحل:

```
x = [24 214 327 617];
y = [375 223 218 341];
plot(x,y)
axis equal
I = imread('picture ');
imshow(I)
hold on
plot(x,y,'r','LineWidth',5)
hold off
```

حيث:

picture : اسم الصورة الموجودة في نفس مسار تشغيل البرنامج.



واجهات المستخدم الرسومية

- التحكم بواجهات المستخدم.
- توابع الاستدعاء.
- بيئة تطوير واجهات المستخدم الرسومية GUIDE.
- تطوير الواجهات الرسومية

مـدخـل إلـى واجـهـات الـمـسـتـخـدم الـرسـومـيـة
Graphic User Interface

١,٦ مقدمة

إن المشكلة التي كانت تواجه مستخدمي الماتلاب قدرته الضعيفة على العرض وإبراز العمل ، ولذلك فإن الكثير من المبرمجين كانوا يعتمدون عليه في الحساب وتنفيذ وحل المشاكل الرياضية ثم يستخدمون نتائجه في برامج أخرى مثل Visual Basic مثلاً .

في النسخ الحديثة من الماتلاب ظهرت وبقوة فكرة إدخال عملية بناء الواجهات في الماتلاب وهو ما يعرف بواجهات المستخدم الرسومية ، إن هذه الواجهات جعلت من برنامج الماتلاب برنامجاً متكاملاً من حيث السهولة في البرمجة وحل المسائل والقدرة على عرض وإظهار النتائج بشكل رائع ، وإنشاء بيئة تفاعلية بينه وبين المستخدم.

٢,٦ مفهوم واجهات المستخدم الرسومية و كيف تعمل

إن واجهات المستخدم الرسومية عبارة عن عرض بياني تخطيطي يتضمن وسائل أو مكونات تؤمن للمستخدم إنجاز مهام فعالة وجذابة ضمن بيئة الماتلاب.

كما إن مكونات الواجهة ممكن أن تكون قوائم (Menus) ، أشرطة الأدوات (Toolbars) ، أزرار الضغط ، أزرار خيار...إلخ ، كما تستطيع أيضاً في بيئة ماتلاب عبر GUI أن تعرض المعطيات على شكل جداول أو كأشكال بيانية ، وتستطيع تجميع المكونات المترابطة .

أما آلية العمل فتعتمد على المبدأ التالي :

كل مكون (جزء) في الـ GUI وحتى الواجهة ككل تترافق بواحدة أو اكثر من الإجرائيات المكتوبة من قبل المستخدم والتي تعرف بالاستدعاءات (Callbacks).

غالباً ما يعرف هذا النوع من البرمجة بالبرمجة المقادة بالحدث (Event-Driven Programming) إن الحدث هنا هو ضغط الزر ، في البرمجة المقادة بالحدث فإن تنفيذ الاستدعاء يكون متزامناً ، ومتحكماً به من قبل أحداث موجهة من خارج البرمجيات.

٣,٦ الكائنات المتاحة في الواجهات

المكون	الوصف
Push Button	تقوم أزرار الضغط بفعل معين عندما ينقر عليها عندما تنقر على زر الضغط ، سيظهر أنه نقر وعندما تحرر الفأرة عنه فإن الزر سيظهر مرتفعاً.
Toggle Button	تولد أزرار التبديل فعلاً وتشير فيما إذا كانت مفعلة أو ملغية ، عندما تضغط على زر التبديل فإنه يظهر مضغوطاً موضحاً أنه مفتوح ،

	عندما تحرر زر الفأرة على عكس زر الضغط فإن زر الضغط يبقى مضغوطاً حتى تنقره مرة أخرى.
Radio Button	إن أزرار الخيار تشبه مربعات الاختيار ولكن تكون نوعاً متعارضة ضمن مجموعة من أزرار الخيار المرتبطة ، أي يمكنك أن تحدد فقط زراً واحداً في نفس الوقت وتتم قيادة هذه الكائنات من قبل Button Group.
Check Box	إن مربعات الاختيار تولد فعلاً عندما تختبر وتشير لحالتها أكانت مختبرة أم غير مختبرة، إن مربعات الاختيار مفيدة عندما يتم تزويد المستخدم بعدد من الاختبارات المستقلة التي تضبط النمط.
Edit text	إن عناصر تحكم النص المحرر عبارة عن حقول تمكن المستخدمين من إدخال أو تعديل العبارات النصية.
Static text	إن عناصر تحكم النص الستاتيكي تعرض خطوطاً من النصوص.
Slider	تقبل المنزلقات دخلاً رقمياً ضمن مجال محدد وذلك بتمكين المستخدم من تحريك الشريط المنزلق الذي يدعى بالمنزلقة أو الإبهام
List Box	تعرض مربعات القائمة قائمة من الأجزاء وتمكن المستخدمين من اختيار أحد الأجزاء أو أكثر.
Pop-Up Menu	تعرض القائمة المنسدلة قائمة من الاختبارات عندما ينقر المستخدمين على السهم.
Axes	إن المحاور تمكن واجهتك من عرض الرسومات والصور.
Panel	تجمع اللوحة مكونات الواجهة ، بالتجميع البصري بالنسبة لأدوات التحكم ، فإذا حركت اللوحة فإن أولادها ستتحرك معها وتحافظ على مكانها على اللوحة.
Button Group	تعتبر مجموعة الأزرار هذه شبيهة باللوحات ولكن تستخدم لتقود سلوك الاختبار الحصري لأزرار الخيار وأزرار التبديل.
ActiveX Component	تمكنك هذه المكونة من عرض أدوات التحكم الفعالة في واجهتك.

٤,٦ برمجة عناصر التحكم

١) Radio Button

إن هذا الزر يمتلك خاصية التحديد أو عدمه فهو إذا يبرمج بالطريقة التالية :

```
if(get(hObject,'Value')==get(hObject,'Max'))
else
end
```

ما إذا كان لدينا عدة أزرار من هذا النوع ونريد تحديد أحداها و إزالة التحديد عن الباقين فإن العبارات البرمجية تكون :

```
Function radiobutton1_callback(hObject,eventdata,handles)
Set([handles.tadiobutton2 handles.radiobutton3],'Value',0)
Function radiobutton2_callback(hObject,eventdata,handles)
Set([handles.tadiobutton1 handles.radiobutton3],'Value',0)
Function radiobutton3_callback(hObject,eventdata,handles)
Set([handles.tadiobutton2 handles.radiobutton1],'Value',0)
```

٢) Pop-Up Menus

نضع خيارات عديدة في هذه القائمة المنسدلة عن طريق String وعندما نريد أن نخص كل خيار ببرنامج معين فإن الvalue تقدم لنا رقم السطر الذي اخترناه في String و هذا يمكننا من كتابة التالي :

```
Function popupmenu_callback(hObject,eventdata,handles)
Val=get(hObject,'Value');
Switch val
case 1
case 2
end
```

ولكن إذا كان لدينا خيارات عديدة ونريد التعامل مع ما كتبناه وليس مع رقم السطر :

```
Function popupmenu_callback(hObject,eventdata,handles)
val=get(hObject,'Value');
String_list=get(hObject,'string');
Selected_string=string_list{val};
```

إن نتيجة string_list ستكون مصفوفة من النوع cell وفيها جميع الخيارات التي وضعت في string.

٣) Toggle Button

إن هذا الزر يأخذ وضعيتين فعند الضغط يأخذ قيمة ١ في الـ value الموجودة في الخصائص وقيمة 0 عن الإفلات :

```
Function togglebutton_callback(hObject,eventdata,handles)
Button_state=get(hObject,'Value');
if  button_state==1      …….
elseif button_state==0       …….
end
```

٤) Check Boxes

```
Function checkbox1_callback(hObject, eventdata, handles)
if(get(hObject,'Value')==get(hObject,'Max'))
else
end
```

٥) edit text

```
Function edittext1_callback(hObject, eventdata, handles)
user_string = get(hObject,'string');
```

٦) Sliders

```
Function slider1_callback(hObject, eventdata, handles)
slider_string = get(hObject,'Value');
```

٧) List Box

إن هذا الزر يشبه Pop-Up Menu إلا أنك تستطيع ان تختار هنا عدة خيارات في نفس الوقت. ضع هذا الكائن في صفحة فارغة من GUI واكتب في String عدة خيارات ثم شغل الملف وحاول أن تختار اكثر من خيار تجد إنك لن تستطيع ذلك فماذا نفعل إزاء ذلك ؟ نقوم بفتح خصائص الزر وتغيير قيمة max إلى قيمة غير الواحد ، ثم نعود ونلاحظ أنه أصبح بإمكاننا اختيار عدة خيارات.

```
Function listbox1_callback(hObject, eventdata, handles)
index_selected = get(handles.listbox1,'Value');
list = get(handles.listbox1,'string');
item_selected = list(index_selected);
```

٥,٦ أهم التعليمات المستخدمة في GUI

تعليمة get :

مهمتها ببساطة أخذ أو معرفة أي خاصية لأي عنصر تريد بمعنى أنه عندما تضغط على أي عنصر مرتين فإن النافذة التي سوف تفتح أمامك هي نافذة خصائص هذا العنصر ، إن كل هذه الخاصيات تستطيع معرفتها عن طريق تعليمة get.

```
Variable=get(handles.buttonname,'property')
```

تعليمة set :

هذه التعليمة مقابلة لعمل get فهي تقوم بوضع أو تغيير خاصية معينة ما برمجياً.

```
set(handles.buttonname,'property',variablename)
```

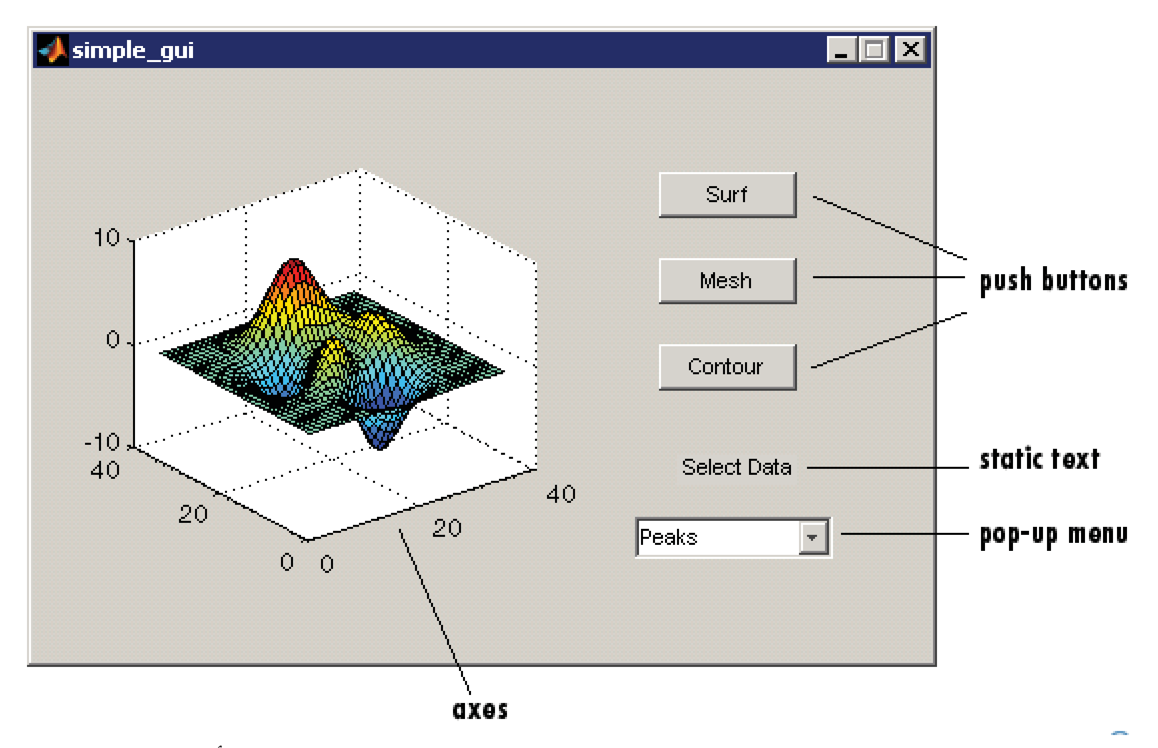
تعليمة global :

ملف البرمجة يحوي على العديد من التوابع لكن هذه التوابع مستقلة أي أن المتغيرات الموجودة في function ما لا تؤثر على المتغيرات الموجودة في الآخر (المتغيرات المحلية) حتى لو كانا يملكان نفس الاسم فلذلك أحياناً نريد تصدير قيمة المتغير من تابع لآخر فنلجأ لجعل هذا المتغير عالمي أو شامل وذلك باستخدام هذه التعليمة.

تصميم الواجهات الرسومية برمجياً
Graphic User Interface

١,٧ إنشاء GUI بسيطة

نوضح فيما يلي كيف يمكنك كتابة البرنامج النصي الذي ينشئ البرنامج النصي الذي واجهة المستخدم GUI الموضحة في الشكل المبين .



إن كافة الكائنات الموجودة في البرنامج المبينة تم التعرف عليها في الدرس الأول ، إن واجهة المثال تتضمن :

- جملة محاور axes.
- قائمة منبثقة pop-up menu تسرد مجموعات مختلفة من المعطيات و التي تقابل توابع في MATLAB هي sine , membrane , peaks.
- نص استاتيكي من أجل وضع تسمية للقائمة المنبثقة.
- ثلاثة أزرار ضغط تعطي ثلاثة أنواع مختلفة من الأشكال contour , mesh , surface.

من أجل استخدام هذا المثال (GUI) فإن المستخدم يختار مجموعة من المعطيات من القائمة المنبثقة ، وعندها فإنه ينقر أحد من أزرار ضغط نوع الرسم ، إن النقر على الزر يشمل تنفيذ الاستدعاء الذي يرسم المعطيات المختارة ضمن جملة المحاور.

٢,٧ أهم التوابع المستخدمة

يوفر برنامج الماتلاب مجموعة من التوابع لإنشاء وبناء واجهات المستخدم الرسومية. يوضح الجدول التالي أهم التوابع التي تحتاجها لبناء المثال المدروس.

التابع	الوصف
Align	يحاذي محاور و عناصر التحكم لواجهة المستخدم
Axes	ينشئ عناصر محاور
Figure	ينشئ عناصر الشكل حيث أن GUI هي عنصر شكل
Movegui	يحرك شكل GUI إلى مكان محدد على الشاشة
Uicontrol	ينشئ عناصر التحكم بواجهة المستخدم مثل أزرار الضغط ، نص ستاتيكي ، و القوائم المنبثقة

٣,٧ إنشاء ملف MATLAB لواجهة المستخدم الرسومية

ابدأ بإنشاء و بناء ملف M-file للواجهة المدروسة .

١. على موجه MATLAB ، اكتب edit فيقوم الماتلاب بفتح المحرر.
٢. اكتب التعليمات التالية. تعليمة التابع هذه هي السطر الأول من الملف function . simple_gui
٣. أضف هذه التعليقات إلى ملف MATLAB بعد تعليمة التابع. سوف تُعرض هذه التعليقات في سطر الأمر استجابة للأمر help. و يجب أن تتبع بسطر فارغ.

```
function simple_gui_test
% SIMPLE_GUI Select a data set from the pop-up menu, then
% click one of the plot-type push buttons. clicking the button
% plots the selected data in the axes

end
```

نحتاج هنا لكتابة التعلمية end بسبب أن المثال مكتوب باستخدام التوابع المتداخلة (nested .(functions

٤. خزن الملف على مجلدك الحالي أو في المكان الذي فيه مسار ملفات MATLAB الخاصة بك. الخطوة التالية ستكون تخطيط الواجهة GUI البسيطة .

٤,٧ تخطيط GUI

يتم تخطيط واجهة المستخدم الرسومية على مرحلتين :

- بناء الشكل .
- إضافة المكونات.

- بناء الشكل (تصميم الواجهة) :

في برنامج الماتلاب ، فإن الواجهة عبارة عن شكل. و بالتالي فإن الخطوة الأولى هي أن تنشئ الشكل و نوضعه على الشاشة. إنها أيضاً تجعل GUI غير مرئية و لذا فإن مستخدم GUI لا يستطيع أن يرى المكونات المضافة أو المهيأة. عندما يتم إضافة جميع المكونات على الواجهة و تكون مهيأة ، فإن المثال سيجعلها مرئية.

```
% Initialize and hide the GUI as it is being consructed.
f=figure('Visible','off','Position',[360,500,450,285]);
```

و يمكن تغيير لون خلفية الواجهة بكتابة :

```
f=figure('Visible','off','Position',[360,500,450,285],'Color',[0.1 0.2 0.6]);
```

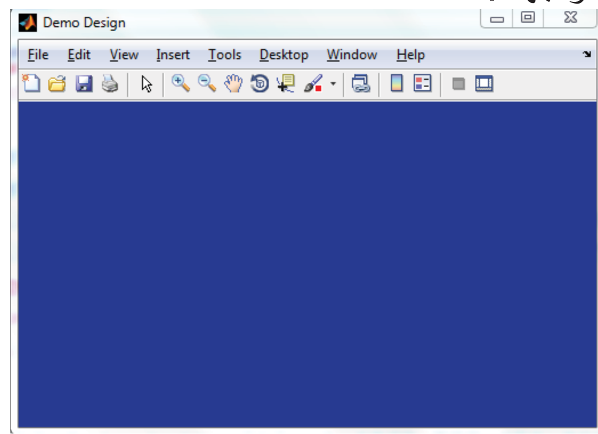
لو أردنا وضع أسماء نختارها نحن لتكون مناسبة أكثر لمحتوى الواجهة الرسومية أو لتحمل اسم التطبيق ، يمكن ببساطة كتابة :

```
f=figure('Visible','off','Position',[360,500,450,285],'Color',[0.1 0.2 0.6],'Name','Demo Design');
```

لو أردنا التخلص من التسمية التلقائية التي تضيفها الواجهة نكتب و ببساطة :

```
f=figure('Visible','off','Position',[360,500,450,285],'Color',[0.1 0.2 0.6],'Name','Demo Design','NumberTitle','off');
```

وبالنتيجة نحصل على الواجهة :



لو أردنا تعطيل صلاحية تغيير حجم الشاشة نكتب :

```
f=figure('Visible','off','Position',[360,500,450,285],'Color',[0.1 0.2 0.6],'Name','Demo Design','NumberTitle','off','Resize','off');
```

والغاية من ذلك الحفاظ على توزع العناصر المكونة للواجهة و عدم إنحصارها في ركن واحد عند تكبيرها. يمكن أن تحدث هذه المشاكل عندما تكون الإحداثيات المسندة للمكونات ثابتة و مستقلة عن حجم الواجهة الذي تحدده قيمة المتغير ScreenSize.
كان التعريف السابق لقيمة المتغير Position بطريقة بدائية . لأنه لم يكن هناك أخذ بعين الاعتبار لحجم الشاشة المتغير من حاسوب لآخر ، ولتجاوز هذا الإشكال يمكننا إستغلال المتغير ScreenSize

```
SCREENSIZE=get(0,'ScreenSize')
f=figure('Visible','off','Position',[SCREENSIZE(1) SCREENSIZE(2) SCREENSIZE(3) SCREENSIZE(4)],'Color',[0.1 0.2 0.6],'Name','Demo Design','NumberTitle','off','Resize','off');
```

إن طلب تابع figure يستخدم زوجي قيمة/خاصية property/value. إن خاصية Position عبارة عن شعاع رباعي العناصر (4-element) يحدد مكان GUI على الشاشة و أبعادها : [المسافة من اليسار ، المسافة من الأسفل ، الارتفاع، العرض]. الواحدات الافتراضية هي البيكسل pixels.
ويمكن للمبرمج إضافة عدة تعديلات أخرى على الواجهة الرسومية عبر إدراج خاصيات أخرى. و يبين الجدول التالي أهم الخصائص و وظيفة كل منها .

الخاصية	الوظيفة
Color	تعديل لون الواجهة
Menubar	إبقاء أو حذف شريط المهام
Name	تسمية الواجهة الرسومية
Numbertitle	تعديل ترقيم الواجهة
Parent	تحديد الواجهة الأم

Position	تحديد المكان و المقاييس
Resize	صلاحية تعديل حجم الشاشة
Tag	تحديد المؤشر
Toolbar	إبقاء أو حذف شريط الأدوات
Userdata	بيانات المستخدم
Visible	إظهار أو إخفاء الواجهة

- إضافة المكونات :

يضم المثال المدروس لواجهة GUI ستة مكونات : ثلاثة أزرار ضغط ، نص ستاتيكي واحد ، وقائمة منبثقة واحدة ، وجملة محاور واحدة. لنبدأ بكتابة التعليمات التي تضيف هذه المكونات إلى GUI. إن إنشاء أزرار الضغط ، النص الستاتيكي، القائمة المنبثقة يتم بالتابع uicontrol. أما إنشاء المحاور فيتم من خلال التابع axes.

١. أضف أزرار الضغط الثلاثة إلى واجهتك بإضافة هذه التعليمات إلى الملف M-file وذلك بعد طلب التابع figure.

```
% Construct the components.
hsurf = uicontrol('style','pushbutton','String','Surf','Position',[315,220,70,25]);
hmesh = uicontrol('style','pushbutton','String','Mesh','Position',[315,180,70,25]);
hcontour = uicontrol('style','pushbutton','String','countour','Position',[315,135,70,25]);
```

تستخدم هذه التعليمات التابع uicontrol لإنشاء أزرار الضغط. كل تعليمة تستخدم سلسلة من الأزواج قيمة/خاصية propert/value لتعريف زر الضغط. ويوضح الجدول و يصف أهم الخصائص مع العلم أنه في كل طلب يتم استرجاع مقبض المكونة التي تم إنشاؤها.

الخاصية	الوصف
Style	في المثال ، pushbutton تحدد المكون كزر ضغط
String	نحدد الاسم الذي سيظهر على كل زر ضغط.
Position	تستخدم شعاع رباعي العناصر لتحديد مكان كل زر ضغط ضمن GUI وقياسه ، و الواحدات الافتراضية هي البيكسل.
Parent	تحديد الواجهة الأم
Backgroundcolor	تحديد لون الخلفية
Tag	تحديد المؤشر
FontWeight	تحديد نوع الكتابة
Callback	الاستدعاء ، تحديد اسم الدالة التي نريد استدعائها لتنفيذ السطور البرمجية المطلوبة عن الضغط على الكائن
Interruptible	التعطيل ، توفر إمكانية تعطيل الوظيفة المسندة للمكون ، كما يوحي إسمها.
Value	تحديد القيمة

٢. أضف القائمة المنبثقة و عنوانها إلى واجهتك بإضافة هذه التعليمات إلى الملف M-file و ذلك بعد تعريفات زر الضغط.

```
hpopup=uicontrol('style','popupmenu','String',{'Peaks','Membrane','Sine'},'Position',[300,50,100,25]);
htext=uicontrol('style','text','String','Select Data','Position',[325,90,60,15]);
```

من أجل القائمة المنبثقة فإن خاصية String تستخدم مصفوفة خلية لتحدد الأجزاء الثلاثة من القائمة المنبثقة : Peaks , Membrane , Sinc . أما بالنسبة للنص الستاتيكي فإنه يستخدم لعنونة القائمة المنبثقة. إن خاصية String لهذا النص تخبر مستخدم GUI ليختار معطيات Select Data. الوحدات الافتراضية لهذه المكونات هي البيكسل.

٣. أضف المحاور إلى GUI وذلك بإضافة التعليمة التالية إلى الملف M-file. اضبط خاصية Units على البيكسل بحيث انها تملك نفس واحدات المكونات الاخرى.

```
ha=axes('Units' , 'pixels' , 'Position',[50,60,200,185]);
```

٤. حاذي جميع المكونات ما عدا المحاور حول مركزها وذلك بالتعليمة التالية ، أضفها إلى الملف M-file بعد كل تعريفات المكونات :

```
align([hsurf,hmesh,hcontour,htext,hpopup],'Center','None');
```

٥. اجعل واجهتك GUI مرئية هذا الأمر بعد أمر المحاذاة :

```
set(f,'Visible','on')
```

٦. بهذا الشكل سيكون ملف M-file كما يلي :

```
function simple_gui
% SIMPLE_GUI Select a data set from the pop-up menu, then
% click one of the plot-type push buttons. clicking the button
% plots the selected data in the axes

% Initialize and hide the GUI as it is being consructed.
f=figure('Visible','off','Position',[360,500,450,285]);

% Construct the components.
hsurf = uicontrol('style','pushbutton','String','Surf','Position',[315,220,70,25]);
hmesh = uicontrol('style','pushbutton','String','Mesh','Position',[315,180,70,25]);
hcontour = uicontrol('style','pushbutton','String','countour','Position',[315,135,70,25]);

hpopup=uicontrol('style','popupmenu','String',{'Peaks','Membrane','Sine'},'Position',[300,50,100,25]);
htext=uicontrol('style','text','String','Select Data','Position',[325,90,60,15]);

ha=axes('Units' , 'pixels' , 'Position',[50,60,200,185]);
align([hsurf,hmesh,hcontour,htext,hpopup],'Center','None');

set(f,'Visible','on')
end
```

تهيئة واجهة المستخدم الرسومية و برمجتها

٥,٧ تهيئة واجهات المستخدم الرسومية GUI

عندما نجعل واجهتك مرئية Visible، فإنها يجب أن تهيأ بحيث تكون جاهزة للمستخدم. سنوضح فيما يلي كيف يمكن :

١. بناء (تصميم) الواجهة الرسومية بحيث تبدو أو تظهر بشكل لائق عندما يضبط قياسها بتغيير واحدات الشكل و المكونة إلى normalized. يؤدي ذلك إلى إعادة ضبط قياس المكونات وجعلها متناسبة مع بعضها عند تغيير ضبط قياس الواجهة GUI. إن الواحدات normalized تعطي الزاوية اليسارية السفلى لنافذة الشكل القيمة (٠،٠) والزاوية اليمينية العلوية القيمة (١،١).
٢. توليد المعطيات للرسم. إن المثال يحتاج ثلاث مجموعات من المعطيات peaks_data , membrane_data , sinc_data. كل مجموعة من هذه المجموعات تقابل أحد أجزاء القائمة المنبثقة.
٣. إنشاء الرسم الأولي على المحاور.
٤. تخصيص اسم الواجهة الذي سيظهر على عنوان النافذة.
٥. تحريك الواجهة الرسومية إلى مركز الشاشة.
٦. جعل الواجهة الرسومية مرئية.

أ) نستبدل الشيفرة التالية في ملف (M-file) :

```
% Make the GUI visible
set(f,'Visible','on')
```

بهذه الشيفرة :

```
% Initialize the GUI
% Change units to normalized so components resize automatically.
set([f,hsurf,hmesh,hcontour,htext,hpopup],'Units','normalized');
% Generate the data to plot.
peaks_data=peaks(35);
membrane_data=membrane;
[x,y]=meshgrid(-8:.5:8);
r=sqrt(x.^2+y.^2);+eps;
sinc_data=sin(r)./r;
% Create a plot in the axes.
current_data=peaks_data;
surf(current_data);
% Assign the GUI a name to appear in the window title.
set(f,'Name','Simple GUI')
% Move the GUI to the center of the screen
movegui(f,'center')
% Make the GUI visible
set(f,'Visible','on')
```

ب) وبالتالي سيبدوا الملف M-file الخاص بواجهتك كما يلي :

جـ) قم بتنفيذ برنامجك النصي عن طريق كتابة simple_gui على سطر الأوامر ، وبالتالي ستظهر واجهتك كما يلي.

```
function simple_gui
    % SIMPLE_GUI Select a data set from the pop-up menu, then
    % click one of the plot-type push buttons. clicking the button
    % plots the selected data in the axes

    % Initialize and hide the GUI as it is being consructed.
    SCREENSIZE=get(0,'ScreenSize')
    f=figure('Visible','off','Position',[360,500,450,285]);

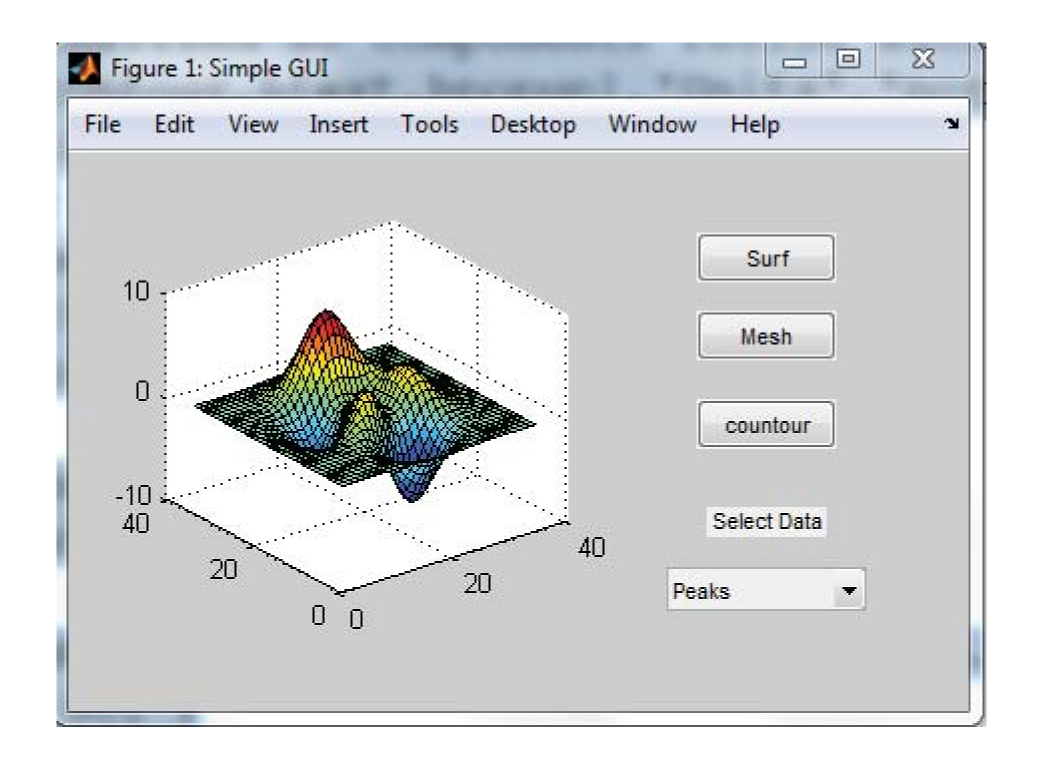
    % Construct the components.
    hsurf = uicontrol('style','pushbutton','String','Surf','Position',[315,220,70,25]);
    hmesh =
    uicontrol('style','pushbutton','String','Mesh','Position',[315,180,70,25]);
    hcontour =
    uicontrol('style','pushbutton','String','countour','Position',[315,135,70,25]);

    hpopup=uicontrol('style','popupmenu','String',{'Peaks','Membrane','Sinc'},'Posit
    ion',[300,50,100,25]);
    htext=uicontrol('style','text','String','Select Data','Position',[325,90,60,15]);

    ha=axes('Units' , 'pixels' , 'Position',[50,60,200,185]);

    align([hsurf,hmesh,hcontour,htext,hpopup],'Center','None');

    % Initialize the GUI
    % Change units to normalized so components resize automatically.
    set([f,hsurf,hmesh,hcontour,htext,hpopup],'Units','normalized');
    % Generate the data to plot.
    peaks_data=peaks(35);
    membrane_data=membrane;
    [x,y]=meshgrid(-8:.5:8);
    r=sqrt(x.^2+y.^2);
    sinc_data=sin(r)./r;
    % Create a plot in the axes.
    current_data=peaks_data;
    surf(current_data);
    % Assign the GUI a name to appear in the window title.
    set(f,'Name','Simple GUI')
    % Move the GUI to the center of the screen
    movegui(f,'center')
    % Make the GUI visible
    set(f,'Visible','on')
    end
```



٦,٧ برمجة واجهة المستخدم الرسومية GUI

إنك ستستخدم استدعاءات لتبرمج مكونات GUI. إن الاستدعاءات عبارة عن توابع يتم تنفيذها بالتجاوب مع الأحداث المولدة من قبل المستخدم مثل نقر الفأرة مثلاً. نوضح فيما يلي كيف يمكن كتابة الاستدعاءات من أول واجهة بسيطة ، و بعدها نبين كيف يمكن ربط كل استدعاء مع مكونته.

- برمجة القائمة المنبثقة في الواجهة الرسومية

إن القائمة المنبثقة تمكن المستخدمين من اختيار المعطيات التي سترسم. عندما يقوم مستخدم GUI باختيار أحد المجموعات الثلاثة من المعطيات ، فإن MATLAB يضبط خاصية Value للقائمة المنبثقة بالدليل المقابل للعبارة النصية المختارة. عندها فإن استدعاء القائمة المنبثقة يقرأ خاصية Value لهذه القائمة ليحدد ما هو الجزء الذي يجب عرضه حالياً و يضبط Current_data تبعاً لذلك.

أضف الاستدعاء التالي إلى ملفك بعد شيفرة التهيئة و قبل تعليمة end النهائية.

```
% Pop-up menu callback. Read the pop-up menu Value property
% to determine which item is currently displayed and make it
% the current data. This callback automatically has access to
% current data because this function is nested at a lower level.
   function popup_menu_Callback(source,eventdata)
      %Determine the selected data set.
      str=get(sourse,'string');
      val=get(source,'Value');
      % Set current data to the selected data set.
```

```
    switch str{val}
       case 'Peaks' %User selects Peaks
          current_data=peaks_data;
       case 'Membrane' %User selects Membrane
          current_data=membrane_data;
       case 'Sinc'  % User selects Sinc
          current_data=sinc_data;
    end
end
```

- برمجة ازرار الضغط في الواجهة الرسومية

ينشئ كل من أزرار الضغط الثلاثة نوعاً من الرسم باستخدام المعطيات المحددة من الاختيار الحالي في القائمة المنبثقة. إن استدعاءات زر الضغط ترسم المعطيات الموجودة في current_data. إنها و بشكل أوتوماتيكي تدخل إلى Current_data بسبب أنها متداخلة عند مستوى منخفض.

أضف الاستدعاءات التالية إلى ملف MATLAB الخاص بك بعد استدعاء القائمة المنبثقة و قبل تعليمة end النهائية :

```
% Push button callbacks. Each callback plots current_data in the
% specified plot type.

   function surfbutton_Callback(source,eventdata)
      % Display surf plot of the currently selected data.
      surf(current_data)
   end

   function meshbutton_Callback(source,eventdata)
      % Display mesh plot of the currently selected data.
      mesh(current_data)
   end

   function contourbutton_Callback(source,eventdata)
      % Display contour plot of the currently selected data.
      contour(current_data)
   end
```

- ربط الاستدعاءات مع مكوناتها

عندما يقوم مستخدم الواجهة الرسومية باختيار مجموعة المعطيات من القائمة المنبثقة أو ينقر على أحد أزرار الضغط ، فإن MATLAB ينفذ الاستدعاء المرافق مع هذا الحدث الجزئي. و لكن كيف يعرف MATLAB ماهو الاستدعاء الذي سينفذ ؟ من أجل ذلك يجب عليك أن تستخدم خاصية callback للمكونات لتحديد اسم الاستدعاء الذي يترافق مع المكونة.

١. لتعليمة uicontrol التي تعرف زر الضغط Surf، أضف زوج قيمة/خاصية property/value التالي :

'Callback',{@surfbutton_Callback}

إن الاستدعاء Callback هو اسم الخاصية ، بينما Surfbutton_callback هو اسم الاستدعاء الذي يخدم في زر الضغط Surf.

٢. بشكل مشابه ، بالنسبة للتعليمة uicontrol التي تعرف القائمة المنبثقة أضف زوج قائمة/خاصية كمايلي :

'Callback',{@popup_menu_Callback}

٧,٧ تنفيذ واجهة المستخدم الرسومية النهائية

من خلال الفقرات السابقة قمت ببناء واجهة بسيطة ، و فيما يلي نبين الملف M-file النهائي و كيفية تنفيذ الواجهة الرسومية. طبعاً الآن أصبح لديك برنامج M-file موضح بالتعليمات التالية :

```
function simple_gui
% SIMPLE_GUI Select a data set from the pop-up menu, then
% click one of the plot-type push buttons. clicking the button
% plots the selected data in the axes

% Initialize and hide the GUI as it is being consructed.
SCREENSIZE=get(0,'ScreenSize')
f=figure('Visible','off','Position',[360,500,450,285]);

% Construct the components.
hsurf =
uicontrol('style','pushbutton','String','Surf','Position',[315,220,70,25],'Callback',{@surf
button_Callback});
hmesh =
uicontrol('style','pushbutton','String','Mesh','Position',[315,180,70,25],'Callback',{@me
shbutton_Callback});
hcontour =
uicontrol('style','pushbutton','String','countour','Position',[315,135,70,25],'Callback',{@
contourbutton_Callback});

hpopup=uicontrol('style','popupmenu','String',{'Peaks','Membrane','Sinc'},'Position',[3
00,50,100,25],'Callback',{@popup_menu_Callback});
htext=uicontrol('style','text','String','Select Data','Position',[325,90,60,15]);

ha=axes('Units' , 'pixels' , 'Position',[50,60,200,185]);

align([hsurf,hmesh,hcontour,htext,hpopup],'Center','None');

% Initialize the GUI
% Change units to normalized so components resize automatically.
set([f,hsurf,hmesh,hcontour,htext,hpopup],'Units','normalized');
% Generate the data to plot.
peaks_data=peaks(35);
membrane_data=membrane;
[x,y]=meshgrid(-8:.5:8);
r=sqrt(x.^2+y.^2);
```

```
sinc_data=sin(r)./r;
% Create a plot in the axes.
current_data=peaks_data;
surf(current_data);
% Assign the GUI a name to appear in the window title.
set(f,'Name','Simple GUI')
% Move the GUI to the center of the screen
movegui(f,'center')
% Make the GUI visible
set(f,'Visible','on')

% Pop-up menu callback. Read the pop-up menu Value property
% to determine which item is currently displayed and make it
% the current data. This callback automatically has access to
% current data because this function is nested at a lower level.

   function popup_menu_Callback(source,eventdata)
      %Determine the selected data set.
      str=get(source,'string');
      val=get(source,'Value');
      % Set current data to the selected data set.
      switch str{val}
         case 'Peaks' %User selects Peaks
            current_data = peaks_data;
         case 'Membrane' %User selects Membrane
            current_data = membrane_data;
         case 'Sinc'  % User selects Sinc
            current_data = sinc_data;
      end
   end

% Push button callbacks. Each callback plots current_data in the
% specified plot type.

   function surfbutton_Callback(source,eventdata)
      % Display surf plot of the currently selected data.
      surf(current_data)
   end

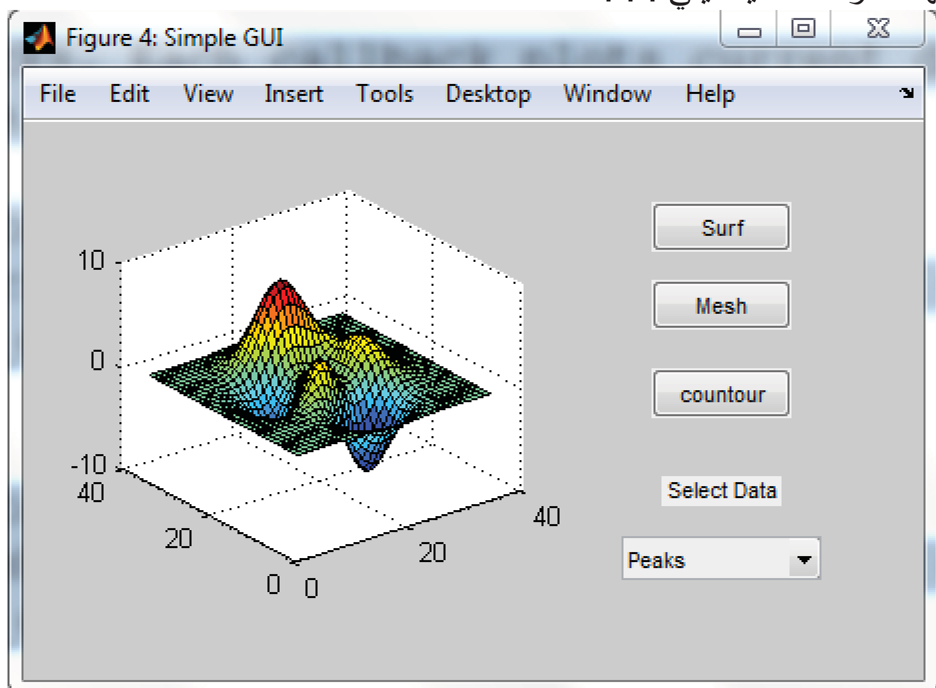
   function meshbutton_Callback(source,eventdata)
      % Display mesh plot of the currently selected data.
      mesh(current_data)
   end

   function contourbutton_Callback(source,eventdata)
      % Display contour plot of the currently selected data.
      contour(current_data)
   end
end
```

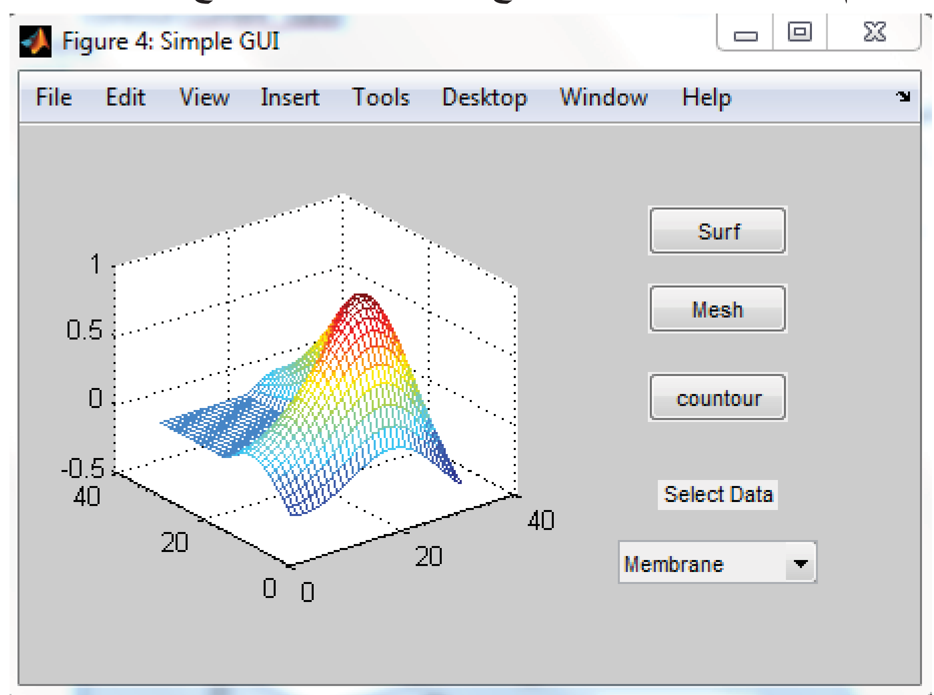
من أجل تنفيذ الواجهة الرسومية قم بالخطوات التالية :

١. نفذ الواجهة الرسومية البسيطة بكتابة اسم ملف M-file على سطر الأوامر
Simple_gui

ستظهر لك الواجهة الموضحة فيما يلي . . .



٢. في القائمة المنبثقة ، اختر Membrane و انقر على الزر Mesh. ستظهر لك الواجهة رسم شعار MATLAB من النوع Mesh كما هو موضح.



٣. قم بتنفيذ كل الخيارات الاخرى الممكنة قبل إغلاق الواجهة GUI.

ما هو GUIDE ؟

١,٨ ما هو GUIDE ؟

إن GUIDE عبارة عن بيئة تطوير واجهة المستخدم الرسومية GUI في MATLAB ، حيث تؤمن هذه البيئة مجموعة من الأدوات لإنشاء واجهة رسم المستخدم GUI. هذه الأدوات تبسط عملية تخطيط و برمجة GUI.

٢,٨ تخطيط واجهة المستخدم الرسومية GUI ؟

باستخدام محرر تخطيط GUIDE، يمكنك أن تملأ الواجهة GUI بالمكونات التي تريد عبر نقر و سحب مكونات GUI مثل الأزرار ، حثول النص ، المنزلقات ، المحاور . . . و . . .، إلى منطقة التخطيط و العرض. يمكنك أيضاً أن تنشئ القوائم و قوائم السياق للواجهة GUI. ومن محرر التخطيط، يمكنك تحديد قياس GUI، تعديل شكل و طبيعة المكونات، محاذاة المكونات، ضبط درجة الجدولة، عرض قائمة هرمية لعناصر المكونات وضبط خيارات GUI.

٣,٨ برمجة الواجهات GUI

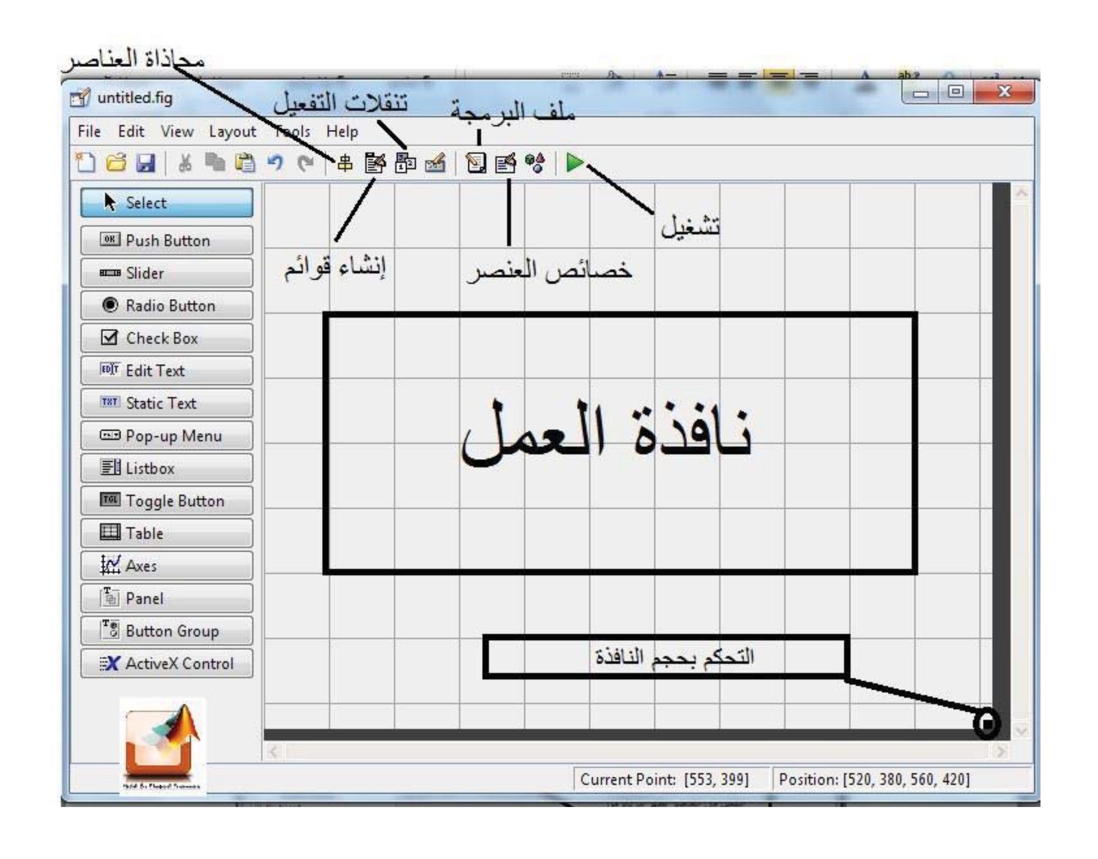
يولد GUIDE بشكل أوتوماتيكي ملف M-file الذي يتحكم بكيفية عمل الواجهة GUI. هذا الملف (M-file) يحوي الشيفرة التي يتم فيها تهيئة الواجهة و يتضمن إطار الاستدعاءات للواجهة GUI (الإجرائيات التي يجب أن تنفذ بالتجاوب مع النقر على مكونات GUI). وبالتالي فإنه يمكنك باستخدام محرر ملف M-file إضافة شيفرة إلى الاستدعاءات لإنجاز المهام التي تريدها.

٤,٨ خلاصة أدوات GUIDE

إن أدوات GUIDE المتاحة في محرر التخطيط مبينة بالشكل.

إن الأدوات المسماة على الشكل موصوفة باختصار في الجدول المرفق. حيث يوضح الجدول كيفية استخدامها في الواجهة.

يمكنك أيضاً ضبط التفضيلات التي تطبق على كل الواجهات التي يتم إنشاؤها و الخيارات التي تخص واجهة محددة.



الأداة (Tool)	استخدام هذه الأداة من أجل :
محرر التخطيط Layout Editor	يختار المكونات من لوحة المكونات الموجودة على الجانب اليساري من المحرر و يرتبها في منطقة التخطيط.
جدولة ضبط قياس الشكل Figure Resize Tab	يضبط القياس للواجهة التي تكون معروضة بشكل أولي.
محرر القائمة Menu Editor	ينشئ قوائم و امتدادات (قوائم منبثقة...).
عناصر المحاذاة Align Objects	تحاذي و تقسم مجموعات المكونات، الشبكات و المساطر تمكنك أيضاً من محاذاة المكونات على الشبكة بالتوصيل الاختياري للشبكة.
محرر درجة الجدولة Tab Order Editor	يضبط درجة الجدولة و التخزين للمكونات في منطقة تخطيطك.
مراقب الخاصية Property Inspector	يحدد خصائص المكونات على مكان التخطيط لواجهتك، حيث يظهر قائمة من كل الخصائص التي يمكنك أن تضعها و يعرض القيم الحالية لها.
مكتشف الهدف Object Browser	يعرض قائمة هرمية من العناصر الموجودة على الشكل.
نفذ Run	يخزن و ينفذ الواجهة الحالية و يعرض الملف M-file المرافق مع الواجهة في محررك الافتراضي.

٥,٨ تفضيلات GUIDE

يمكنك ضبط التفضيلات لمحرر تخطيط GUIDE باختيار Preferences من قائمة File في الماتلاب. هذه التفضيلات تطبق على كل الواجهات التي تقوم بإنشائها.

إن التفضيلات تكون موجودة بأماكن مختلفة ضمن مربع حوار التفضيلات :

- تفضيلات التأكيد.
- خيار التوافق مع الإصدارات الأقدم.
- كل التفضيلات الأخرى.

- تفضيلات التأكيد

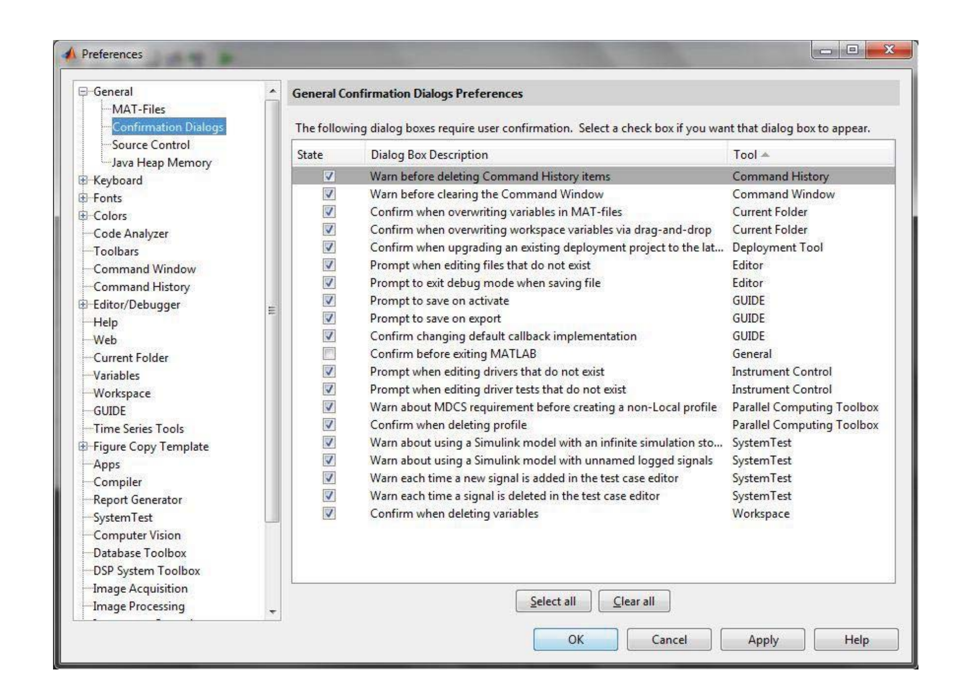
يوجد في محرر تخطيط GUIDE مربعي حوار يستخدمان للتأكيد، حيث يمكنك أن تختار ماذا تريد أن يعرض مربع حوار التأكيد عندما تقوم بما يلي :

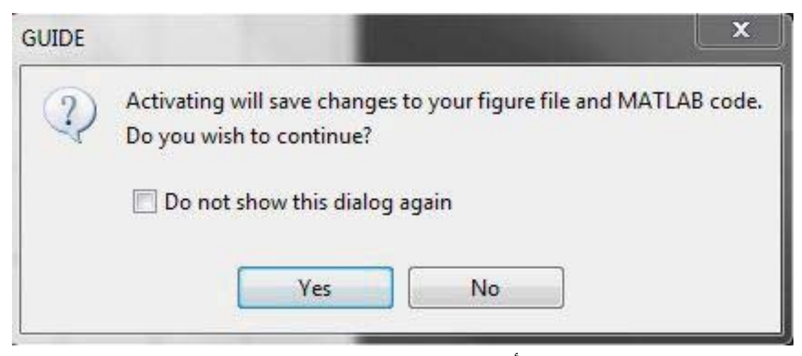
- تفعِل الواجهة GUI.
- تصدِر الواجهة GUI.

في مربع حوار تفضيلات MATLAB، انقر General -> Confirmation Dialogs للدخول إلى تفضيلات التأكيد لمحرر التخطيط ، كما بالشكل ، ابحث عن كلمة GUIDE في العمود Tool.

أ. خيار التخزين مع التفعيل :

عندما تقوم بتفعيل واجهتك GUI بالنقر على زر Run، سيظهر لك مربع حوار ليخبرك عن التخزين الوشيك و يدعك تختار فيها إذا كنت تريد أو لا تريد الاستمرار وذلك كما يوضح الشكل.

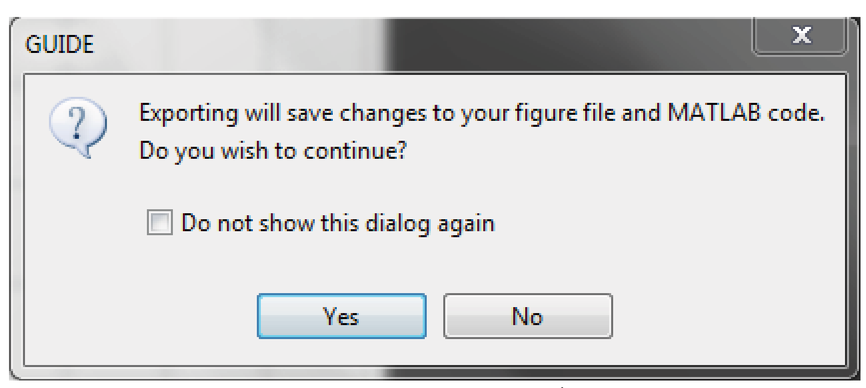




شاشة تأكيد التخزين - للتنفيذ

ب. خيار التخزين مع التصدير :

عندما تقوم باختيار Export من قائمة File لمحرر التخطيط، سيظهر لك مربع حوار يخبرك عن التخزين الوشيك و يجعلك تختار فيما إذا كنت تريد الاستمرار أم لا، كما بالشكل .

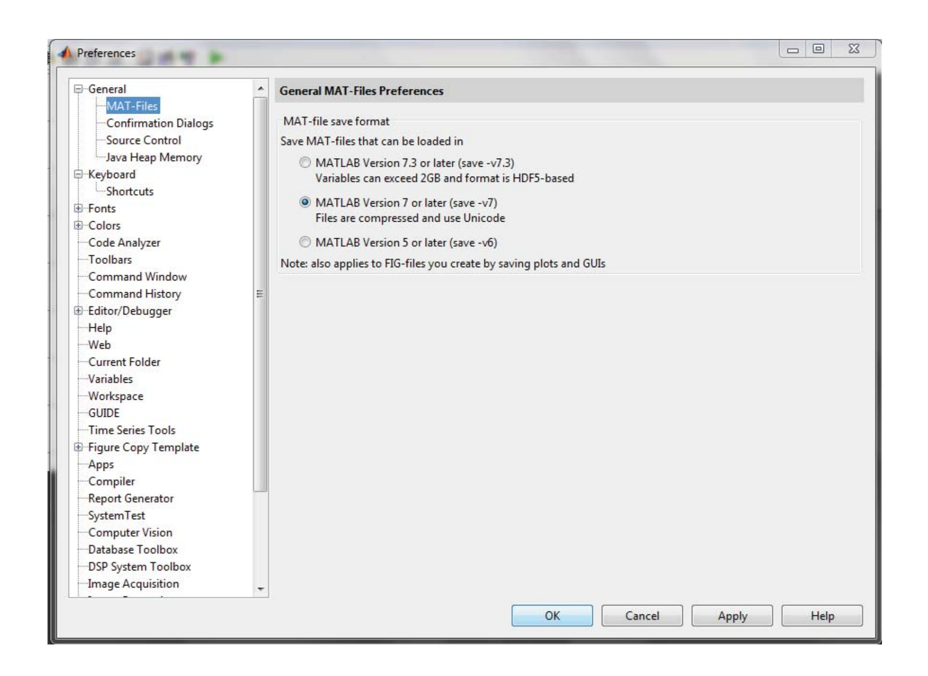


شاشة تأكيد التخزين - للتصدير

- خيار التوافق مع الإصدارات الأقدم

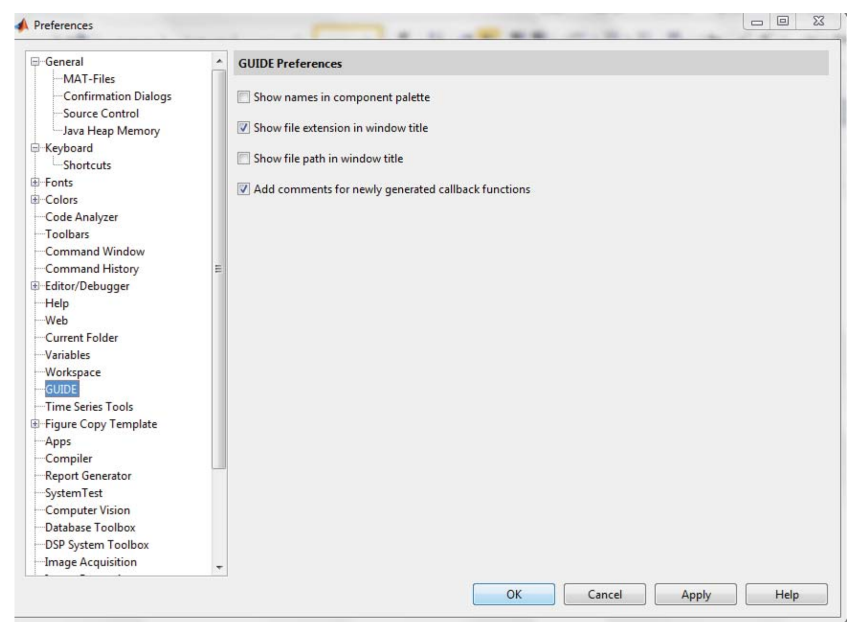
إن ملفات الشكل FIG-Files للواجهات الرسومية التي تم إنشاؤها أو تطويرها مع النسخة MATLAB 7.0 أو النسخ الأحدث من ذلك، تكون غير متوافقة بشكل أوتوماتيكي مع النسخة MATLAB 6.5 والنسخ الأقدم. لجعل ملف الشكل FIG-File الذي يكون نوع من الملفات MAT، متوافق مع النسخ الأقدم، فإنه يجب عليك أن تختبر التفضيل : Ensure backward compatibility (-v6) وذلك النسبة للملف MAT-file و الذي يكون في مربع حوار Preferences تحت MAT-Files كما هو موضح بالشكل.

ملاحظة : الواجهات الرسومية التي تم تطويرها مع النسخة MATLAB 7.3 والأحدث لا تكون متوافقة مع النسخ MATLAB 7.2 والتي قبلها و يجب الأخذ بعين الاعتبار لذلك عند حفظ ملف الواجهة الرسومية.



- التفضيلات الأخرى

إن محرر تخطيط GUIDE مزود بتفضيلات أخرى متعددة. في مربع حوار تفضيلات MATLAB، انقر GUIDE للدخول إلى التفضيلات الباقية لمحرر التخطيط كما هو موضح.



كما هو موضح بالشكل فإن التفضيلات تقوم بما يلي :

أ. عرض أسماء في لوحة المكونات.

ب. عرض امتداد الملف في عنوان النافذة.

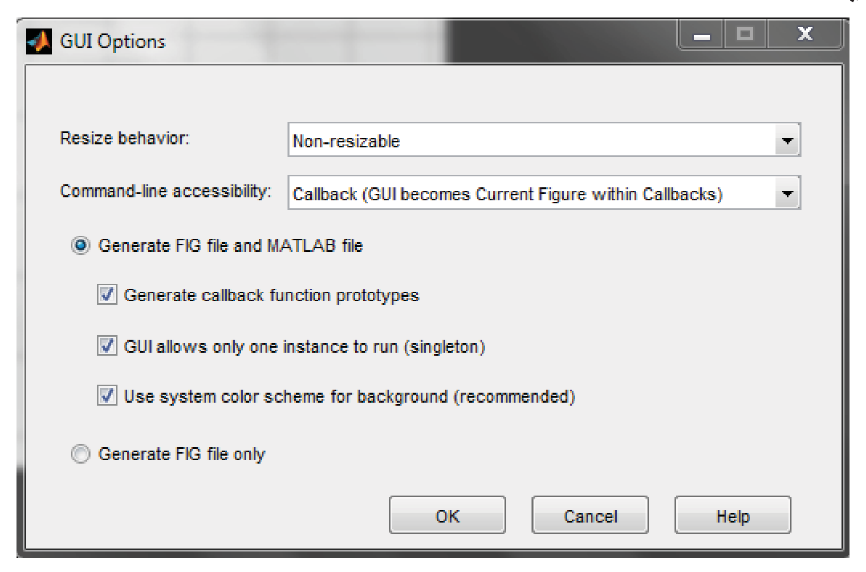
ت. عرض مسار الملف في عنوان النافذة.

ث. إضافة تعليقات لتوابع الاستدعاء المولدة حديثاً.

٦,٨ خيارات واجهة المستخدم الرسومية GUI

بعد فتح قالب جديد للواجهة GUI في محرر التخطيط، وقبل أن تخزن هذه الواجهة، يمكنك تكوين خيارات سير عمل مختلفة للواجهة و ذلك باستخدام مربع الحوار GUI options. هذه الخيارات يتم تطبيقها على واجهة محددة فقط و هي التي قمت بإنشائها.
إن الدخول إلى مربع الحوار يتم باختيار GUI options من قائمة Tools لمحرر التخطيط، فيظهر لك كما هو مبين الخيارات التالية :

- سلوك ضبط القياس.
- قابلية الدخول إلى سطر الأمر.
- توليد ملف FIG-File و M-file.
- توليد ملف شكل FIG-File فقط.



- سلوك ضبط القياس

يمكنك أن تتحكم فيما إذا كان المستخدم يستطيع أن يضبط قياس نافذة الشكل المتضمنة واجهتك و كيف يتعامل MATLAB مع ضبط القياس. إن GUIDE يوفر ثلاث خيارات :

- Non-resizable : لا يستطيع المستخدم في هذه الحالة تغيير قياس النافذة (حالة افتراضية).
- Proportional : يقوم MATLAB في هذا الخيار و بشكل أوتوماتيكي بضبط قياس مكونات الواجهة بشكل نسبي مع قياس نافذة الشكل الجديد.
- Other (يستخدم ResizeFcn) : تتم برمجة الواجهة بحيث يتم ضبط المكونات بطريقة معينة عندما يقوم المستخدم بإعادة ضبط قياس الشكل.

إن الطريقة الأولى و الثانية ببساطة تحدد الخواص بشكل مناسب و لا تتطلب أي إجراء آخر. أما الطريقة الثالثة تتطلب منك أن تكتب إجرائية الاستدعاء التي تعيد حساب قياس و مكان المكونات بناء على قياس الشكل الجديد.

- محاذاة المكونات

إن محرر التخطيط يزودك بعدد من الخصائص التي تسهل محاذاة وتوزيع المكونات بالنسبة لبعضها البعض.

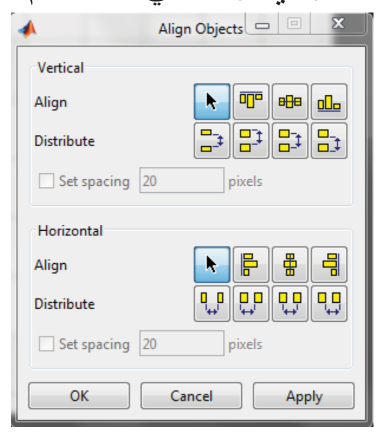
○ أداة المحاذاة

تمكنك أداة المحاذاة من وضع العناصر بشكل متناسق بالنسبة لبعضها البعض وضبط الفراغ بين العناصر المختارة. إن عمليات المحاذاة المحددة تطبق على جميع المكونات التي تم اختيارها عندما تطبق الزر Apply.

إن أداة المحاذاة تزودك بنوعين من عميات المحاذاة :

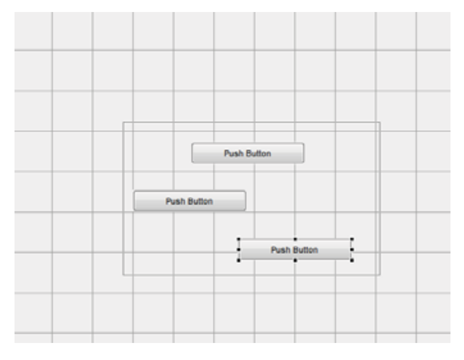
- Align : تحاذي المكونات المختارة إلى سطر مرجعي وحيد.
- Distribute : تضع مساحة لكل المكونات المختارة بشكل تكون متساوية لبعضها البعض.

وكلا النوعين من المحاذاة يمكن أن يطبق في الاتجاهين العمودي والأفقي كما هو في الشكل. لاحظ أنه في عدد من الحالات، فإنه من الأفضل أن نطبق المحاذاة بشكل مستقل بالنسبة للاتجاه العمودي أو الأفقي باستخدام خطوتين منفصلتين.



١. خيارات المحاذاة

هناك نوعين للمحاذاة بشكل أفقي وبشكل عمودي. كل خيار يحاذي المكونات المختارة إلى سطر مرجعي، الذي يتم تحديده بمربع التحديد الذي يضم العناصر المختارة. على سبيل المثال، فإن الصورة في الشكل التالي لمنطقة التخطيط توضح مربع التحديد مشكل من ثلاثة أزرار ضغط مختارة.



إن كل خيارات المحاذاة (أعلى عمودياً، مركزي، يسار أفقي و سفلي، مركزي، يميني) تخطط المكونات المختارة بالنسبة للطرف المقابل لمربع التحديد المختار.

٢. خيارات التوزيع

إن توزيع المكونات يضيف فراغات متساوية بين كل المكونات في المجموعة المنتقاة. تعمل خيارات التوزيع وفق نمطين مختلفين :

- فراغات متساوية بين المكونات المختارة ضمن مربع التحديد (الحالة الافتراضية لهذا الخيار).
- تفصل المكونات المختارة بقيمة محددة من البيكسل (اختر مربع الخيار set spacing و حدد قيمة البيكسل).

إن كلا النمطين يمكنك من تحديد قياس الفراغ، كما هو مشار إلى التسميات السفلية لأداة المحاذاة. تتضمن هذه الخيارات فراغات مقاسة بالنسبة للحواف التالية:

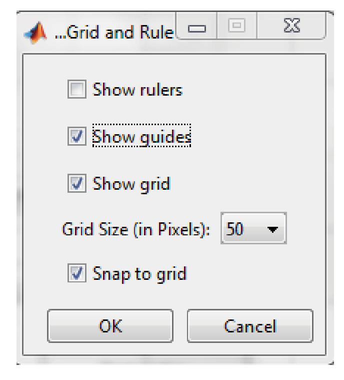
- عمودياً (داخلية، فوق، مركز، وأسفل).
- أفقياً (داخلية، يسار، مركز، ويمين).

- الشبكة و المساطر

إن منطقة التخطيط تعرض شبكة ومساطر لتسهل تخطيط المكونات، خطوط الشبكة مفصولة بمجالات ٥٠ بيكسل بشكل افتراضي ويمكنك أن تختار عدداً من القيم الأخرى تتراوح بين (١٠-٢٠٠) بيكسل. تستطيع بشكل اختياري تمكين خاصية snap-to-grid والتي تسبب تحركاً لأي عنصر أو يعاد ضبط قياسه إلى ضمن ٩ بيكسل من خط الشبكة للقفز إلى هذا الخط. إن خيار snap-to-grid يعمل مع أو بدون الشبكة المرئية.

استخدم مربع حوار الشبكة والمساطر الموضح بالشكل التالي من أجل :

- التحكم برؤية الشبكة والمساطر وخطوط الشبكة.
- ضبط فراغات الشبكة.
- تمكين أو عدم تمكين خاصية snap-to-grid.



- ضبط المرتبة الجدولية

إن المرتبة الجدولية للواجهة GUI عبارة عن تسلسل مكونات الواجهة الرسومية المختارة عندما يقوم المستخدم بنقر المفتاح Tab من لوحة المفاتيح.
توفر أشكال MATLAB عدة مجموعات جزئية منفصلة تتحكم بجدولة الأنواع المختلفة للمكونات:

- عناصر تحكم واجهة المستخدم مثل الأزرار، المنزلقات، والقوائم المنبثقة.
- اللوحات، مجموعة الأزرار button group (المحاور تكون بنفس المجموعة الجزئية ولكن لا تجدول).
- العناصر ActiveX(لا يمكن جدولتها).

يمكنك أن تتحكم بالدرجة الجدولية للمكونات فقط إذا كانت هذه المكونات بنفس المجموعات الجزئية. تحدد المرتبة الجدولية لكل مستوى بشكل مستقل، بحيث يكون الشكل المستوى الأساسي، وكل لوحة أو مجموعة أزرار button group تنشئ مستواها الخاص بها. إذاً، أثناء جدولة المكونات على مستوى الشكل، قام المستخدم بجدولة لوحة أو مجموعة الأزرار button group، عندها يتتابع تسلسل الجدولة ضمن مكونات اللوحة أو مجموعة الأزرار button group قبل العودة إلى المستوى الذي وصلت إليه اللوحة ومجموعة الأزرار button group.
من أجل تغيير مرتبة الجدولة، اختر المكونة وانقر على السهم أعلى up وأسفل down من أجل تحريك المكونة للأسفل أو للأعلى في القائمة.

برمجة واجهة المستخدم الرسومية GUI

٧,٨ الاستدعاءات

عند الإنتهاء من تخطيط الواجهة لابد من برمجة سلوك هذه الواجهة ، إذ تتحكم الشيفرة التي تكتبها بالكيفية التي ستتجاوب بها الواجهة GUI مع الأحداث مثل نقرة زر ، حركة منزلقة ، اختيار جزء من قائمة ، أو إنشاء أو حذف المكونات ، تأخذ هذه البرمجة أشكالاً عدة مثل مجموعة توابع ، طلب الاستدعاءات callbacks ولكل مكونة وحتى للواجهة GUI نفسها.

١. ماهو الاستدعاء Callback :

إن الاستدعاء عبارة عن التابع الذي تكتبه وترفقه مع مكونة GUI محددة أو مع شكل الواجهة GUI، إنه يتحكم بالواجهة GUI أو سلوك المكونة عن طريق إنجاز بعض الأفعال بالتجاوب مع الحدث المطبق على المكونة ، يدعى هذا النوع من البرمجة غالباً بالبرمجة المقادة بالحدث event -driven Programming.
عندما يحدث حدث للمكونة ، فإن الماتلاب يستحضر استدعاء المكونة التي طلب في الحديث ، ويمكن أن تكون المكونة أي وسيلة تحكم مثل زر ضغط أو مربع قائمة أو منزلقة كما يمكن أن تكون أيضاً قائمة أو لوحة أو مجموعة أزرار.

٢. أنواع الاستدعاءات:

إن شكل الواجهة وكل نوع من المكونات يقبل أنواعاً محددة من الاستدعاءات والتي يمكن إرفاقها معها. والاستدعاءات التي تكون متاحة لأي مكونة تكون معرفة بخواص هذه المكونة.

يملك كل نوع من الاستدعاءات آلية تنفيذ أو حدث يسبب استدعاءه والجدول التالي يسرد الخواص الاستدعائية التي يتحها GUIDE مع أحداث تنفيذها والمكونات التي تطبق عليها.

خاصية الاستدعاء	حدث التنفيذ	المكونات التي تستخدم الخاصية
ButtonDawnFcn	تتنفذ عندما يقوم المستخدم بنقر زر بينما يكون المؤشر على أو ضمن ٥ بيكسل من المكونة أو الشكل	المحور – الشكل – مجموعة الأزرار – لوحة – عناصر تحكم واجهة المستخدم
Callback	هو فعل التحكم، وينفذ على سبيل المثال عندما ينقر المستخدم على زر الضغط أو يختار جزءاً من القائمة	قائمة المحتويات – قائمة – عناصر تحكم واجهة المستخدم
CloseRequestFcn	ينفذ عندما يغلق الشكل	الشكل
CreatFcn	يقوم بتهيئة المكونة عندما يتم إنشاؤها، إنه ينفذ بعد إنشاء المكونة أو الشكل ولكن قبل عرضها	المحاور – الشكل – مجموعة الأزرار – اللوحة – قائمة المحتويات – القائمة – عناصر تحكم واجهة المستخدم
DeleteFcn	ينجز عمليات التنظيف فقط قبل إلغاء المكونة أو الشكل	المحاور – الشكل – مجموعة الأزرار – اللوحة – قائمة المحتويات – القائمة – عناصر تحكم واجهة المستخدم
KeyPressFcn	ينفذ هذا الاستدعاء عندما ينقر المستخدم على مفتاح من لوحة المفاتيح ومكونة الاستدعاءات أو الشكل تكون مفعلة	الشكل – عناصر تحكم واجهة المستخدم
ResizeFcn	ينفذ عندما يعيد المستخدم ضبط قياس اللوحة ، مجموعة الأزرار ، أو الشكل حيث تكون خاصية	مجموعة الأزرار – الشكل – اللوحة

	Resize للشكل مضبوطة على on	
SelectionChangeFcn	ينفذ عندما يختار المستخدم زر خيار أو زر تبديل مختلف في مكونة مجموعة الأزرار	مجموعة الأزرار
WindowButtonDawnFcn	ينفذ عندما تضغط زر الفأرة عندما يكون المؤشر ضمن الشكل	الشكل
WindowButtonMotionFcn	ينفذ عندما تحرك المؤشر ضمن نافذة الشكل	الشكل
WindowButtonUpFcn	ينفذ عندما تحرر زر الفأرة	الشكل



إن عناصر التحكم التي تتفاعل مع المستخدم تتضمن أزرار الضغط، المنزلقات، أزرار خيار، مربعات اختيار، مربعات نص قابل للتحرير، مربعات نص ستاتيكي، مربعات قائمة، وأزرار تبديل، يشار إليها أحياناً بعناصر تحكم واجهة المستخدم UIControls.

٨,٨ ملفات واجهة المستخدم الرسومية GUI

١. الملفات M-files و ملفات الشكل FIG-file

عندما تقوم بتخزين أو تنفيذ واجهتك لأول مرة فإن GUIDE سيخزن الواجهة بشكل افتراضي بملفين:

- ملف الشكل FIG-file ذو الامتداد fig. والذي يتضمن وصفاً كاملاً لتخطيط الواجهة و مكونات الواجهة مثل أزرار الضغط، المحاور، اللوحات، القوائم، و إلى ما هنالك. إن ملف الشكل FIG-file عبارة عن ملف رقمي ثنائي ولا يمكن تعديله باستثناء تغيير التخطيط في GUIDE. لاحظ أن هذا الملف FIG-file هو نوع من ملفات MAT-file.
- الملف M-file ذو الامتداد m. والذي بشكل أولي يتضمن شيفرة التهيئة و قوالب لبعض الاستدعاءات التي تكون ضرورية للتحكم بسلوك الواجهة. إذ يجب عليك أن تضيف الاستدعاءات التي تكتبها لمكونات واجهتك إلى هذا الملف.

حالما تخزن واجهتك فإن GUIDE سيفتح بشكل أوتوماتيكي ملف M-file في محرر الافتراضي. إن كلا الملفين FIG-file و M-file يتوضعان عادة بنفس المجلد، وهما يقابلان مهام تخطيط و برمجة الواجهة GUI. عندما تقوم بتخطيط الواجهة من محرر التخطيط، فإن عملك سيتم تخزينه في ملف FIG-file، وعندما تقوم ببرمجة الواجهة، فإن عملك سيتم تخزينه في الملف M-file.



إذا تضمنت الواجهة عناصر ActiveX، فإن GUIDE يولد أيضاً ملفاً لكل مكونة ActiveX.

٢. بنية ملف M-file الواجهة GUI

إن الملف M-file للواجهة GUI الذي يولده GUIDE عبارة عن ملف تابع. إن اسم التابع الأساسي هو نفس الاسم لملف M-file. مثلاً: إذا كان اسم ملف M-file هو mygui.m، عندها فإن اسم التابع الأساسي هو mygui. إن كل استدعاء في الملف هو تابع جزئي Subfunction للتابع الأساسي.

عندما يقوم GUIDE بتوليد ملف M-file، فإنه و بشكل أوتوماتيكي يضم قوالب أغلب الاستدعاءات شائعة الاستخدام لكل مكونة. إن الملف M-file يتضمن أيضاً شيفرة التهيئة، وكذلك استدعاء تابع الافتتاح واستدعاء تابع الخرج. يجب عليك أن تضيف شيفرة إلى استدعاء المكونات لواجهتك من أجل أن تنجز الأعمال التي تريد، ومن جهة أخرى تستطيع أيضاً إضافة شيفرة لاستدعاء تابع الافتتاح واستدعاء تابع الخرج. إن الأجزاء الأساسية لملف M-file الواجهة GUI كما هو في الجدول.

المقطع	الوصف
تعليقات Comments	تعرض على سطر استجابة للأمر help ، إن تحرير هذه التعليقات ضروري لواجهتك
تهيئة Initialization	مهام تهيئة GUIDE، لا تحرر هذه الشيفرة
تابع الافتتاح (opening function)	ينجز مهام التهيئة التي تريد قبل أن يدخل المستخدم إلى الواجهة GUI
تابع الخرج (Output function)	يرجع المخارج إلى سطر أمر ماتلاب بعدما يقوم تابع الافتتاح بإرجاع التحكم وقبل تسليم التحكم إلى سطر الأمر
استدعاءات الشكل والمكونات Component and figure) (callbacks	يتحكم بسلوك شكل الواجهة والمكونات الفردية، إن ماتلاب يطلب الاستدعاء بالتجاوب مع الحدث الجزئي للمكونة أو للشكل نفسه

٩,٨ صياغة اسم الاستدعاء و وسطائه

إن GUIDE يعرف اصطلاحات لأسماء الاستدعاء ووسطائه و يصيغ هذه المصطلحات في قوالب و إصدارات الاستدعاء التي تضاف إلى ملف M-file. كل قال يكون مشابه للتابع الجزئي callback لزر الضغط الموضح في العبارات التالية:

```
% --- Executes on button press in pushbutton١.
function pushbutton١_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

إن سطر التعليق الأول يصف الحدث الذي يشغل تنفيذ الاستدعاء. إن هذا السطر يتبع بسطر تعريف التابع، أما التعليقات الباقية تصف وسطاء الدخل. أدخل شيفرة البرنامج بعد التعليق الأخير.



يمكنك تجنب التوليد الآلي لأسطر تعليقات الاستدعاء للاستدعاءات الجديدة. في مربع حوار التفضيلات Preferences، اختر GUIDE وألغ الخيار:
Add Comments for newly generated Callback functions

١. تسمية توابع الاستدعاء

إن مثال الاستدعاء الموضح أعلاه يوضح التعريف التالي للتابع:

```
function pushbutton١_Callback(hObject, eventdata, handles)
```

عندما يقوم GUIDE بتوليد القالب، فإنه ينشئ اسم الاستدعاء بإلحاق محرف الخط السفلي (_) واسم خاصية الاستدعاء إلى خاصية Tag للمكونة. في المثال المدروس، فإن pushbutton1 تكون خاصية Tag لزر الضغط، وcallback يكون واحداً من خواص الاستدعاء لزر الضغط. إن خاصية Tag تعرف وبشكل وحيد المكونة ضمن الواجهة GUI. في المرة الأولى التي تخزن فيها الواجهة GUI بعد إضافة المكونة، يضيف GUIDE استدعاءات هذه المكونة إلى ملف M-file ويولد أسماء الاستدعاء باستخدام القيمة الحالية لخاصية Tag. وإذا أردت تغيير قيمة Tag الافتراضية فإنه عليك أن تفعل ذلك قبل تخزين الواجهة GUI.

٢. تغيير أسماء الاستدعاءات المخصصة من GUIDE

يمكنك تغيير أسماء الاستدعاءات المخصصة من GUIDE بإحدى الطرق التالية:

أ. تغيير خاصية Tag

يمكنك أن تغير خصائص Tag لإعطاء استدعاءات المكونة اسماً ذا معنى أقوى. أي أنه من الممكن تغيير خاصية Tag من pushbutton1 إلى closebutton. إذا أمكن، قم بتغيير خاصية Tag قبل تخزين الواجهة، وعندها فإن GUIDE وبشكل أوتوماتيكي يستخدم القيمة الجديدة عندما يسمي الاستدعاءات. مع ذلك فإنه إذا قمت بتغيير خاصية Tag بعد تخزين الواجهة GUI، فإن GUIDE سيحدث الأجزاء التالية تبعاً للقيمة الجديدة للـTag ، مؤمناً جعل كل المكونات تملك قيم Tag مختلفة:

- توابع استدعاء المكونات في ملف M-file.
- قيم خصائص استدعاء المكونات، التي تستطيع عرضها في مراقب الخاصية.
- الإشارة في ملف M-file إلى حقل بنية handles التي تتضمن مقبض المكونات.

ب. تغيير خاصية الاستدعاء

لإعادة تسمية تابع جزئي لاستدعاء جزئي بدون تغيير خاصية Tag:

- استبدل العبارة النصية للاسم في خاصية الاستدعاء باسم جديد. على سبيل المثال، إذا كانت قيمة خاصية الاستدعاء لزر الضغط في mygui هي:

```
mygui('pb1_Callback',gcbo,[ ],guidata(hObject))
```

فإن العبارة النصية pb1_Callback عبارة عن الاسم لتابع الاستدعاء. غير الاسم إلى الاسم المطلوب، على سبيل المثال، closebutton.

- يجب عليك أن تحدث كل حالات اسم تابع الاستدعاء في ملف M-file.
- يجب عليك تحديث المراجع في ملف M-file إلى حقل بنية handles التي تتضمن مقبض المكونات.

٣. وسطاء الدخل

إن جميع الاستدعاءات في ملف للواجهة GUI تملك وسطاء الدخل التالية:

- hobject: مقبض العنصر (المكونة) الذي يتم به تنفيذ الاستدعاء.
- eventdata: تحجز لاستخدامات لاحقة.
- handles: بنية تتضمن مقابض العناصر في الشكل. وكذلك معطيات التطبيقات المعرفة.

٤. البنية Handles

ينشئ GUIDE البنية handles التي تتضمن مقابض كل العناصر في الشكل، من أجل الواجهات GUI التي تتضمن نص محرر، لوحة، قائمة منبثقة، وزر ضغط، فإن البنية handles تبدو في الأصل مشابهة للنص التالي، إن GUIDE يستخدم خاصية tag للمكونات لتسمية عنصر البنية وذلك من أجل التعامل معه.

```
handles =
            Figure1: 160.0011
            Edit1: 9.0020
            Uipanel1: 8.0017
            Popupmenu: 7.0018
            Pushbutton1: 161.0011
            Output: 160.0011
```

إن GUIDE ينشئ ويعرض بنية handles كمعطيات للواجهة GUI، حيث تمرر كوسيط دخل لكل الاستدعاءات وتمكن استدعاءات الواجهة من مشاركة قيم الخاصية ومعطيات التطبيق.

١٠,٨ استدعاءات التهيئة

إن GUIDE بشكل تلقائي يتضمن استدعاءين، تابع الافتتاح وتابع الخرج في كل ملف M-file للواجهة GUI التي يتم إنشاؤها وبناؤها.

▪ تابع الافتتاح Opening Function

إن تابع الافتتاح هو الاستدعاء الأول في كل ملف M-file للواجهة. إن هذا الاستدعاء يتم تنفيذه فقط قبل أن تصبح الواجهة GUI مرئية للمستخدم، ولكن بعد أن يتم إنشاء كل المكونات أي أنه بعد أن يتم تنفيذ استدعاءات CreateFcn للمكونات، إذا كان أي منها بحاجة لتنفيذ.

يمكنك استخدام تابع الافتتاح لإنجاز مهام التهيئة قبل أن يدخل المستخدم إلى الواجهة. على سبيل المثال، يمكنك أن تستخدمه لإنشاء المعطيات أو أن تقرأ المعطيات من مصدر خارجي، كما أن وسطاء سطر أمر الواجهة GUI تمرر إلى تابع الافتتاح.

أ. تسمية التابع وقالبه

إن GUIDE يسمي تابع الافتتاح بإلحاقه العبارة OpeningFcn إلى ملف M-file. يقدم النص التالي مثالاً لقالب افتتاح لملف M-file الواجهة mygui:

```
% --- Executes just before mygui is made visible.
function mygui_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in b future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to mygui (see VARARGIN)

% Choose default command line output for mygui
handles.output = hObject;
% Update handles structure
guidata(hObject, handles);
% UIWAIT makes mygui wait for user response (see UIRESUME)
% uiwait(handles.mygui);
```

ب. وسطاء الدخل

إن تابع الافتتاح يملك أربعة وسطاء دخل: varargin، handles، eventdata، وhobject. إن تابع الافتتاح يستطيع أن يصنع مثل هذه الوسطاء المتاحة للاستدعاءات بإضافتها إلى بنية handles.

تمرر وسطاء سطر الأمر إلى تابع الافتتاح خلال varargin. إذا قمت بفتح الواجهة GUI مع الزوج اسم الخاصية/قيمة الخاصية كوسطاء، فإن الواجهة GUI تفتح بالخاصية الموضوعة إلى القيمة المخصصة. على سبيل المثال: my_gui('Position',[71.8 44.9 74.8 19.7]) يفتح الواجهة GUI في المكان المخصص. حيث أن Position هي خاصية شكل قانونية (أي أنها إحدى خواص الشكل القياسية). إذا كان وسيط الدخل ليس خاصية شكل قانونية فإنه يجب عليك أن تضيف شيفرة لتابع الافتتاح لكي تتمكن من استخدام هذا الوسيط.

ت. شيفرة القالب الأولي

بشكل أولي، فإن قالب تابع الدخل يتضمن هذه الأسطر من الشيفرة:

- Handles.output = hObject: يضيف عنصر جديد output إلى البنية handles و يخصصها بقيمة الدخل hObject الذي يكون مقبض الشكل (مقبض الواجهة GUI). إن المقبض سيستخدم لاحقاً من قبل تابع الخرج.
- guidata (hObject, handles) : يخزن البنية Handles. يجب عليك أن تستخدم guidata لتخزين أي تغيرات تنفذها على البنية handles. حيث أنه غير كافي فقط أن تضع القيمة في حقل handles.
- uiwait (handles, mygui) : يكون هذا الخيار معلقاً في البداية، ويقوم بحجب تنفيذ الواجهة حتى يتم استدعاء uiresume أو يتم حذف الواجهة. لاحظ أن uiwait يسمح للمستخدم بالدخول إلى النوافذ الأخرى في الماتلاب.

▪ تابع الخرج

إن تابع الخرج يرجع إلى سطر الأمر MATLAB قيم الخرج (بارامترات الخرج) التي تم توليدها خلال التنفيذ. إن هذا التابع ينفذ عندما يرجع تابع الافتتاح عنصر تحكم وقبل أن يعود عنصر التحكم إلى سطر الأمر. إن هذا يعني أنه يجب أن تولد قيم الخرج في تابع الافتتاح، أو نستدعي uiwait في تابع الافتتاح لإيقاف تنفيذه بينما تقوم الاستدعاءات الأخرى بتوليد قيم الخرج.

أ. تسمية التابع وقالبه

إن GUIDE يسمي تابع الخرج بإلحاقه العبارة _OutputFcn إلى اسم ملف M-file. هذا مثال عن قالب لتابع خرج كما يبدو في ملف mygui M-file:

```
% --- Outputs from this function are returned to the command line.
function varargout = program23_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see
VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in b future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;
```

ب. وسطاء الدخل

إن تابع الخرج يملك ثلاثة وسطاء دخل: handles، eventdata ، hObject .

ت. وسطاء الخرج

إن تابع الخرج يملك وسيط خرج واحد وهو varargout الذي يعود إلى سطر الأمر. بشكل افتراضي فإن تابع الخرج يخصص handles.output للوسيط varargout. هكذا فإن الخرج الافتراضي يكون مقبض الواجهة GUI التي تخصصه العبارة handles.output في تابع الافتتاح. يمكنك تغيير الخرج عن طريق:

- تغيير قيم handles.output. والتي يمكن أن تكون أي قيمة MATLAB قانونية متضمنة بنية أو مصفوفة خلية.
- إضافة وسطاء خرج إلى varargout.

إن varargout عبارة عن مصفوفة خلية. إنها تستطيع أن تتضمن أي عدد من وسطاء الخرج. بشكل افتراضي، فإن GUIDE ينشئ وسيط خرج واحد، handles.output . لإنشاء وسيط خرج إضافي، أنشئ حقلاً جديداً في البنية handles وأضفه إلى Varargout باستخدام الأمر المشابه كمايلي:

varargout{٢} = handles.output;

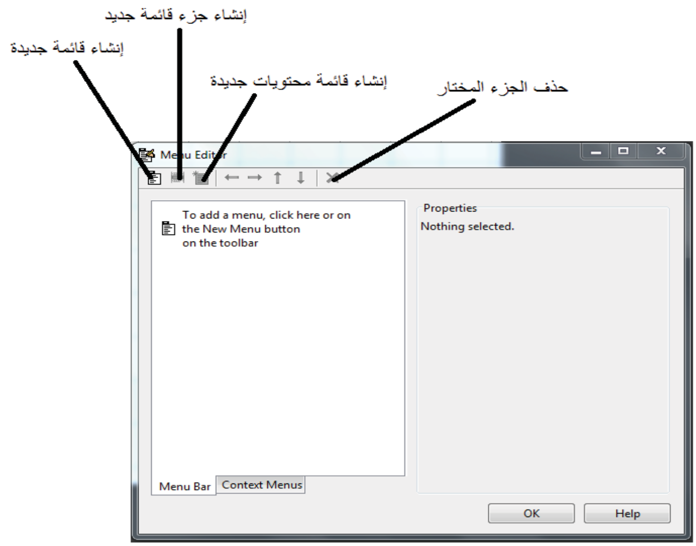
برمجة القوائم و أشرطة الأدوات

١١,٨ إنشاء القوائم

إن GUIDE يمكنك من إنشاء نوعين من القوائم :

- قوائم لشريط القوائم .
- قوائم المحتويات .

يمكنك أن تبني كلا النوعين من القوائم باستخدام محرر القائمة Menu Editor، حيث يتم الدخول إلي محرر القائمة من قائمة Tools أو من شريط أدوات محرر التخطيط. يوضح الشكل التالي شاشة محرر القائمة.

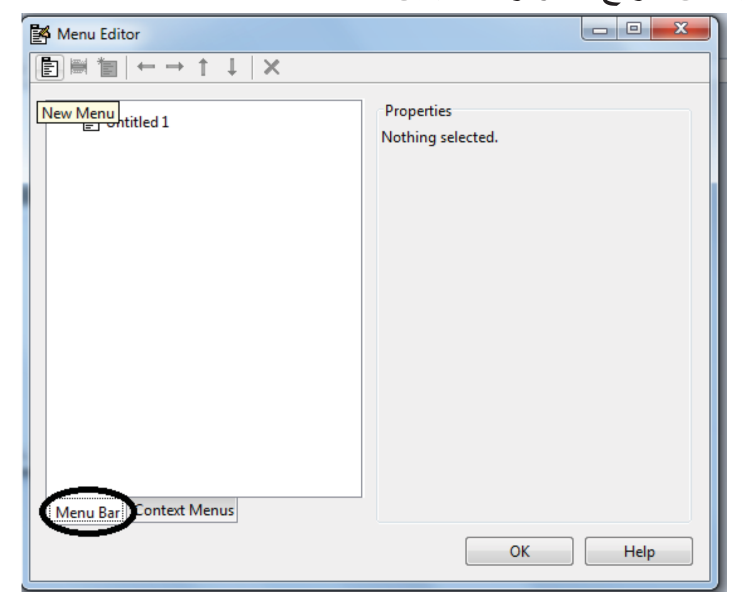


❖ قوائم شريط القائمة

عندما تقوم ببناء قائمة منسدلة فإن GUIDE يضيف عنوان هذه القائمة إلى شريط قائمة GUI عندها فإنه يمكنك إنشاء أجزاء قائمة لهذه القائمة. وكل جزء قائمة يستطيع أن يملك قائمة متسلسلة، تعرف أيضاً بالقائمة الجزئية Submenu. وهذه الأجزاء يمكنها أيضاً أن تملك قائمة متسلسلة و هكذا. عندما تريد بناء القائمة المنسدلة أولاً، فإن GUIDE وبشكل أوتوماتيكي يضيف شريط القائمة إلى الواجهة GUI. ويمكنك عندها أن تنفذ الخيارات التالية :

١. إنشاء القائمة

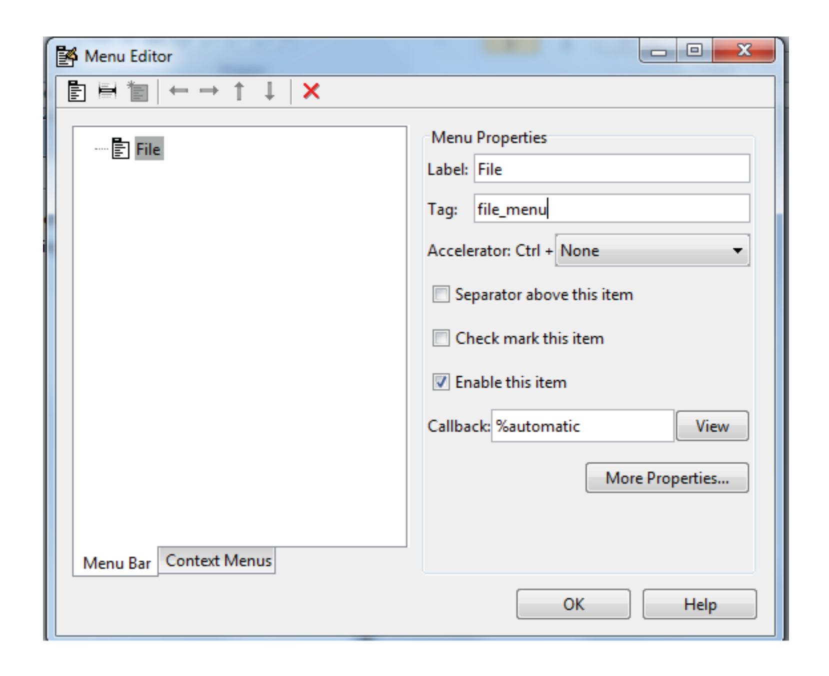
أ. ابدأ بقائمة جديدة و ذلك بالنقر على أداة New Menu. سيظهر عنوان القائمة في الجهة اليسارية من مربع الحوار كما يبين الشكل.



ب. انقر على عنوان القائمة لعرض اختيار خواص القائمة في الجهة اليمينية فيظهر لك الشكل الموضح.

ت. املأ حقلي Label & Tag للقائمة. على سبيل المقال ، حدد Label إلى 'file' وحدد Tag إلى 'file_menu'. انقر خارج الحقل لتطبيق التغييرات التي أدخلتها.

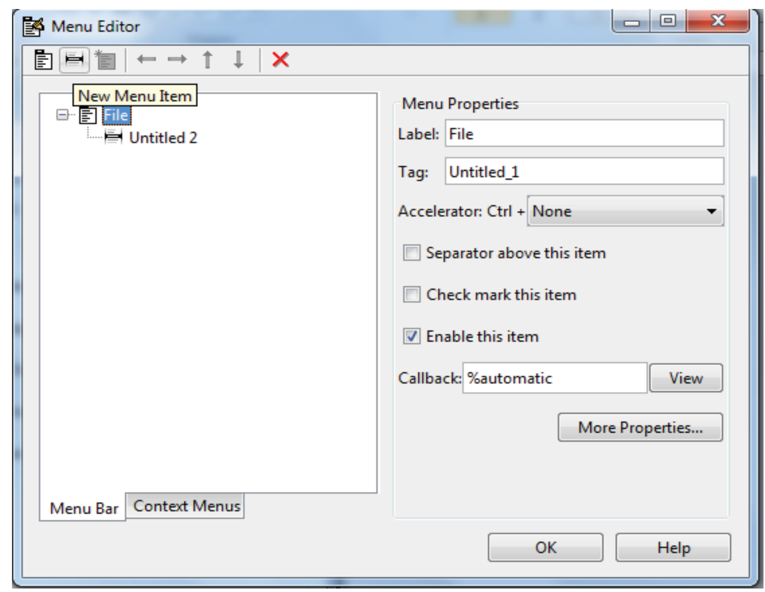
إن العنوان Label عبارة عن عبارة نصية تطلق على عنوان النص لجزء القائمة. و لعرض الحرف '&' في العنوان استخدم حرفين '&&' في العبارة النصية. إن استخدام الكلمتين default & remove (حالة حساسة) ممنوعة، لاستخدام إحداها كتسمية ضع خط مائل (backslash (\)) كسابقة قبل الكلمة. على سبيل المثال ، '\remove' يقود إلى remove. إن الحقل Tag هو عبارة نصية تعرف عنصر القائمة. حيث يستخدم في شيفرة البرنامج لتعريف جزء القائمة و يجب أن يكون وحيداً في الواجهة GUI.



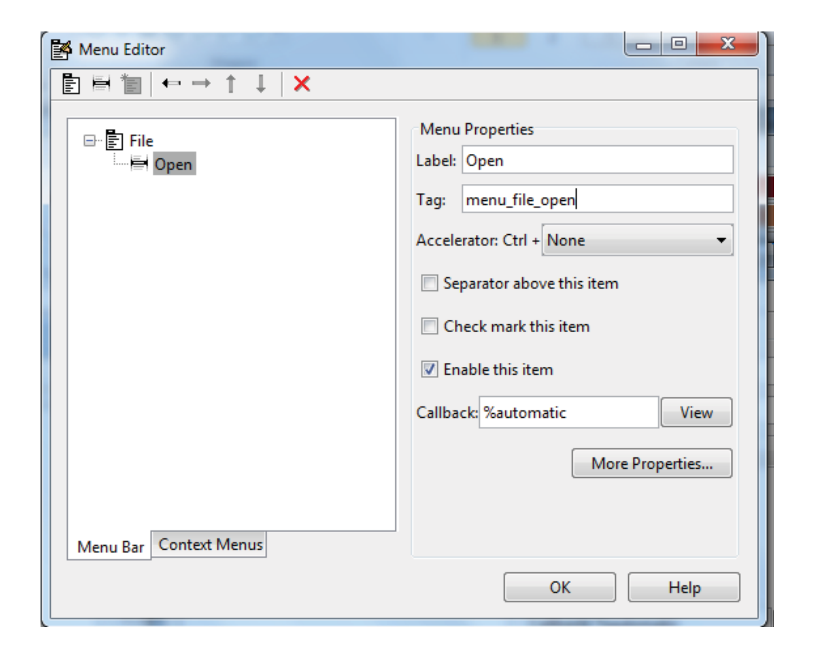
٢. إضافة أجزاء إلى القائمة

استخدم أداة New Menu Item من أجل إنشاء أجزاء القائمة التي سيتم عرضها في القائمة المنسدلة.

أ. أضف جزء القائمة open تحت file وذلك باختيار File و النقر على الأداة New Menu Item. سيظهر عندها عنوان جزء قائمة مرقع مؤقت 'Untitled 2' كما هو موضح.



ب. ضع الحقلين Label و Tag أجزاء القائمة الجديدة ، على سبيل المثال ، ضع 'open' في Label وضع 'menu_file_open' في Tag، و انقر خارج الحقل من أجل تطبيق التغييرات التي نفذتها. عندها ستصبح القائمة كما في الشكل.



يمكنك أيضاً :

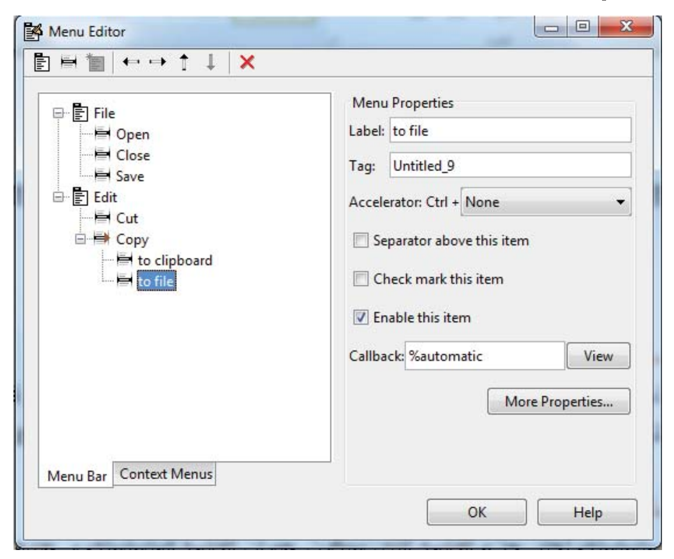
- اختيار مسرع لوحة المفاتيح لجزء القائمة المنبثقة Accelerator. ويكون ذلك بربط أحد المفاتيح مع ctrl. فإن هذا الاختيار سيكون مناسباً لجزء القائمة الذي لا يملك قائمة جزئية. بعض المسرعات يمكن أن تستخدم لأغراض أخرى على نظامك و يمكن أن تعطي و تنتج أفعالاً أخرى.
- اعرض فاصل فوق جزء القائمة باختيار Separate above this item.
- اعرض اختياراً يلي جزء القائمة عندما يتم فتح القائمة أولاً و ذلك باختيار check mark this item، حيث يشير الاختبار إلى الحالة اللحظية لجزء القائمة.
- تمكين هذا الجزء عندما تفتح هذه القائمة لأول مرة و ذلك باختيار Enable this item. يسمح للمستخدمين باختيار هذا الجزء عندما تفتح القائمة لأول مرة. إذا لم تختر هذا الخيار فإن جزء القائمة يظهر فاهياً عند فتح القائمة لأول مرة، و لا يستطيع المستخدم اختياره.
- حدد عبارة نصية لإجرائية التنفيذ (الاستدعاء Callback) الذي ينجز الفعل المرافق مع جزء القائمة. إذا لم تقم بتخزين الواجهة GUI فإن القيمة الافتراضية هي '%automatic'. عندما تخزن الواجهة GUI، ولم تغير هذا الحقل فإن GUIDE و بشكل أوتوماتيكي يضبط القيمة باستخدام تركيب من الحقل Tag واسم ملف GUI. يعرض الزر View الاستدعاء، إذا كان موجوداً في المحرر. و إذا لم تخزن الواجهة GUI فإن GUIDE يسمح لك أن تخزنها.
- افتح مراقب الخاصية ، حيث يمكنك تغيير كل خصائص القائمة بالنقر على زر More options.

٣. قوائم منسدلة إضافية

من أجل إنشاء قوائم منسدلة إضافية ، استخدم الأداة New Menu بنفس الطريقة التي قمت فيها ببناء القائمة File. على سبيل المثال ، يوضح الشكل التالي قائمة منسدلة جديدة Edit.

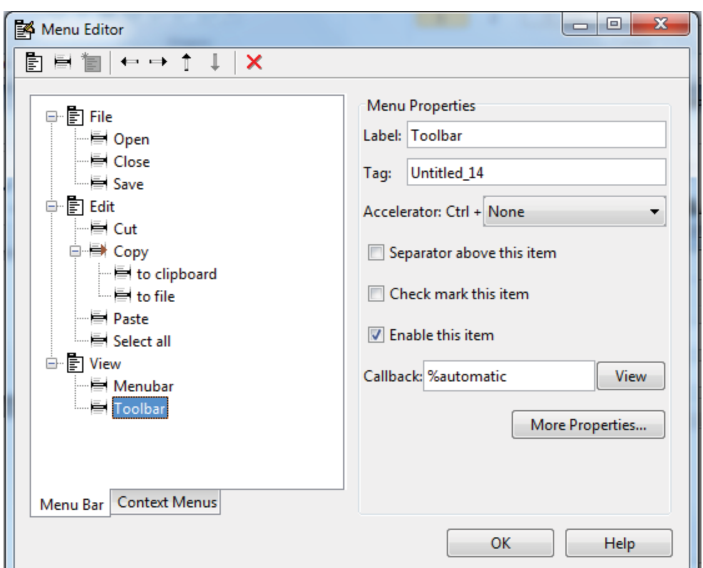
٤. قوائم متسلسلة

لإنشاء قائمة متسلسلة، اختر جزء القائمة الذي سيكون عنوان القائمة المتسلسلة، و عندها انقر الأداة New Menu Item. في المثال الموضح التالي في الشكل فإن جزء القائمة Copy عبارة عن قائمة متسلسلة.



٥. تخطيط ثلاث القوائم

إن محرر التخطيط الموضوح في الشكل يوضح تخطيط ثلاث قوائم معرفة لشريط قائمة الشكل. عندما تقوم بتنفيذ الواجهة GUI، فإن عناوين القوائم ستظهر في شريط القائمة كما هو موضح بالشكل.



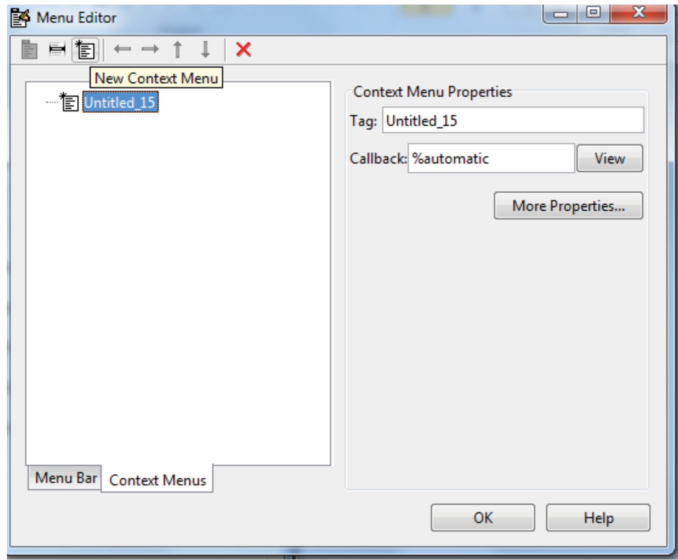
❖ قوائم المحتويات

يتم عرض قائمة المحتويات عندما ينقر المستخدم يميناً على العنصر الذي تكون فيه القائمة معرفة. يمكنك محرر القائمة من تعريف قوائم المحتويات و تضمينها مع العناصر في التخطيط.

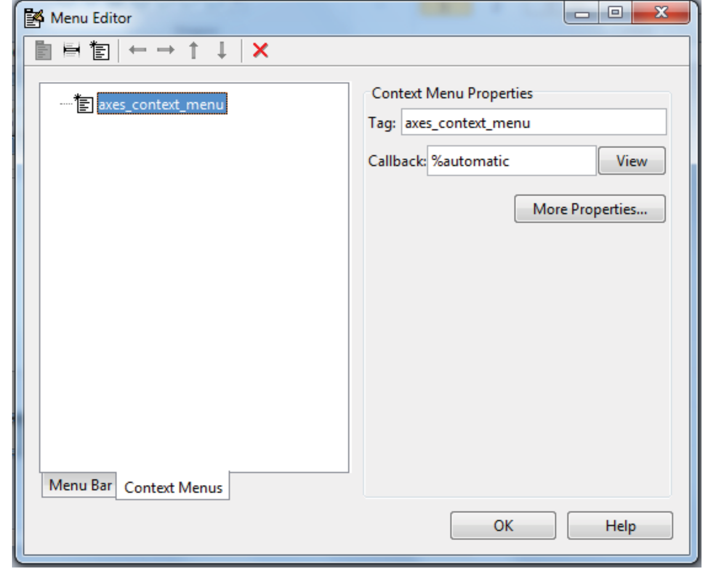
١. إنشاء القائمة الأم

إن كل الأجزاء في قائمة المحتويات تكون أبناء للقائمة التي لا تعرض على شريط قائمة الشكل. لتعريف قائمة الأم قم بما يلي :

أ. اختر قوائم المحتويات Context Menus من محرر القائمة كما في الشكل و اختر قائمة جديدة New Context Menus من شريط الأدوات.



ب. اختر القائمة و حدد Tag لتعرف قائمة المحتويات (في المثال : axes_context_menu) كما في الشكل.

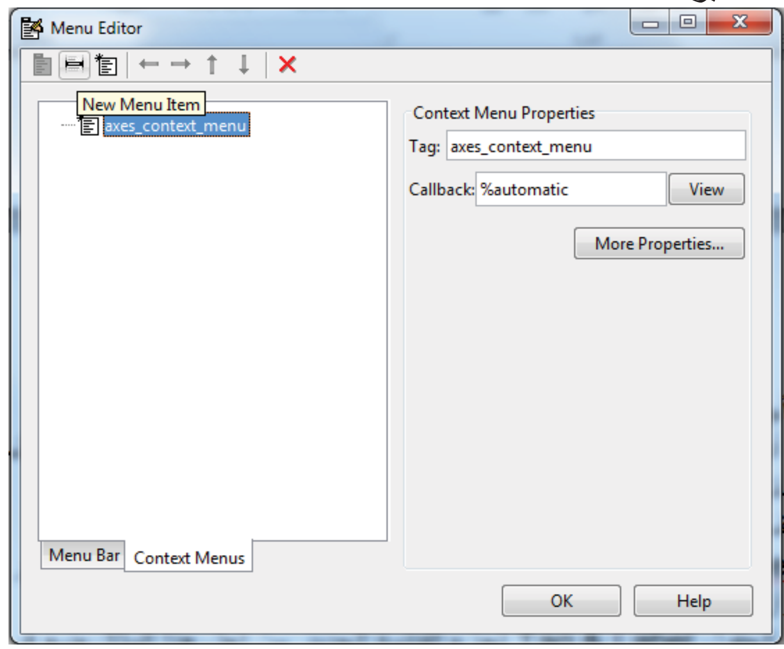


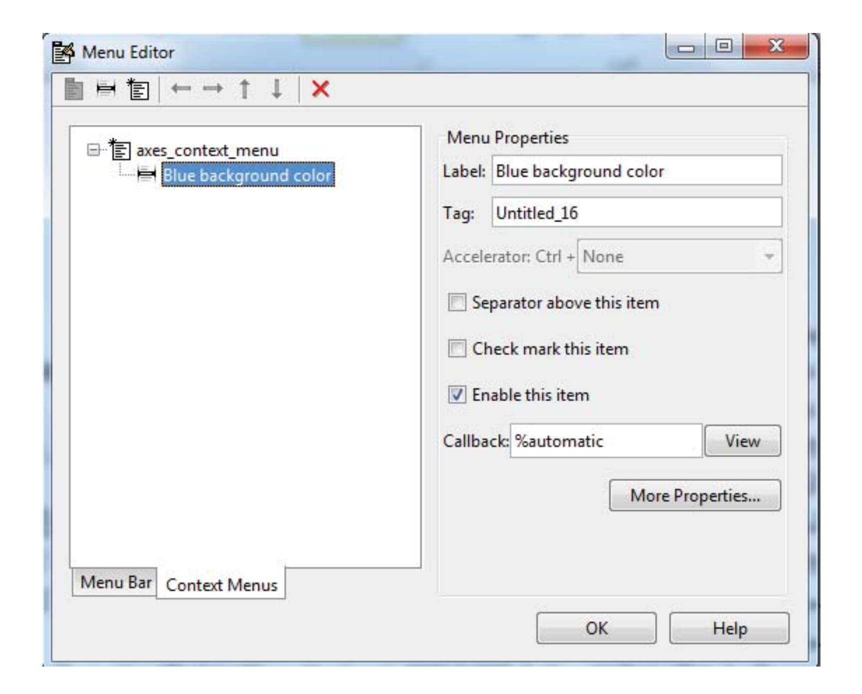
٢. إضافة أجزاء إلى قائمة المحتويات

استخدم أداة New Menu Item لإنشاء أجزاء القائمة التي سيتم عرضها في قائمة المحتويات.

أ. أضف جزء قائمة Blue background color إلى جزء القائمة باختيار axes_context_menu و النقر على أداة New Menu Item، سيظهر عنوان جزء قائمة مرقم مؤقت 'Untitled' كما هو بالشكل.

ب. املأ الحقول Tag & Label لجزء القائمة الجديد. من أجل هذا المثال ضع في Label العبارة 'blue background color' وضع في Tag العبارة : 'blue background' و انقر خارج الحقل من أجل تطبيق التغييرات التي نفذتها. سيظهر لك الشكل الأخير.





يمكنك أيضاً أن تقوم بما يلي :

- عرض فاصل فوق جزء القائمة باختيار Separator above this item.
- عرض اختبار يلي جزء القائمة عندما يتم فتح القائمة أولاً و ذلك باختيار check mark this item. الاختبار حيث يشير إلى الحالة اللحظية لجزء القائمة.
- تمكين هذا الجزء عندما تفتح هذه القائمة لأول مرة وذلك باختيار Enable this item. مما يسمح للمستخدمين باختيار هذا الجزء عندما تفتح القائمة لأول مرة. إذا لم تختر هذا الخيار، فإن جزء القائمة يظهر فاهياً عند فتح القائمة لأول مرة و لا يستطيع المستخدم اختياره.
- حدد العبارة النصية لإجرائية التنفيذ (الاستدعاء callback) التي تنجز الفعل المرافق لجزء القائمة. إذا لم تقم بتخزين الواجهة GUI فإن القيمة الافتراضية هي '%automatic'. عندما تخزن الواجهة GUI، ولم تغير هذا الحقل، فإن GUIDE و بشكل أوتوماتيكي يضبط القيمة باستخدام التركيب المكون من الحقل Tag واسم ملف GUI. إن الزر View يعرض الاستدعاء إذا كان موجوداً في المحرر، إذا لم تخزن الواجهة GUI فإن GUIDE يسمح لك أن تخزنها.
- افتح مراقب الخاصية، حيث يمكنك تغيير كل خصائص القائمة بالنقر على زر More options.

٣. مشاركة قائمة المحتويات مع العنصر

في محرر التخطيط، اختر العنصر الذي تقوم بتعريف قائمة محتويات له ثم استخدم مراقب الخاصية لضبط خاصية العنصر UIContextMenu إلى اسم قائمة المحتويات المطلوبة. يوضح الشكل خاصية UIContextMenu من أجل عنصر محاور axes بخاصية Tag هي axes1. في ملف M-file للواجهة GUI، أكمل التوابع الفرعية للاستدعاء من أجل كل جزء في قائمة المحتويات. كل استدعاء ينفذ عندما يختار المستخدم جزء قائمة المحتويات المقابل.

Property	Value
Tag	axes1
TickDir	in
TickDirMode	auto
TickLength	[0.01; 0.025]
⊞ TightInset	[4.4 1.308 1 0.615]
UIContextMenu	axes_context_menu
Units	<None>
UserData	axes_context_menu
⊞ View	[0.0 90.0]
Visible	on

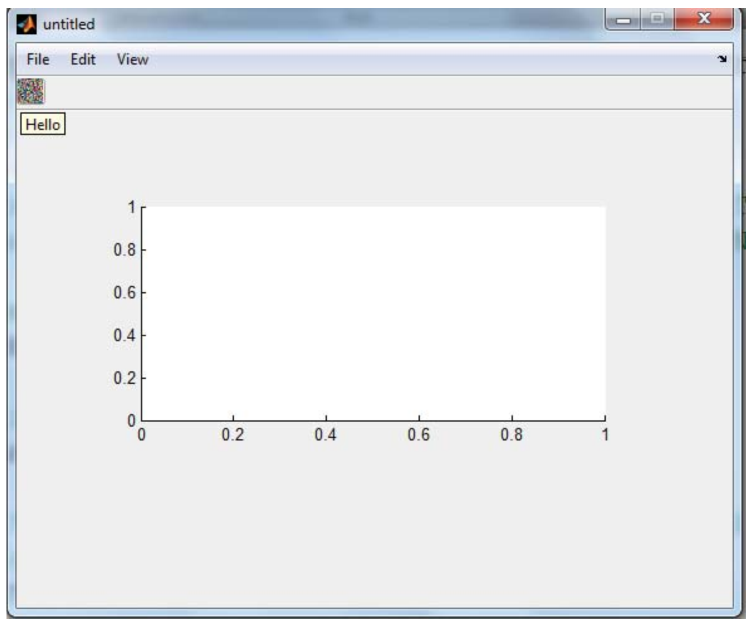
١٢,٨ إنشاء و بناء أشرطة الأدوات

إن GUIDE في كافة إصدارات الماتلاب قبل النسخة MATLAB 2012b لا يزودك بأداة تمكنك من إضافة شريط أدوات، مع ذلك فإنك تستطيع إضافة شريط أدوات بإضافة شيفرة لتابع الافتتاح opening function. أضف الشيفرة التالية إلى تابع افتتاح الواجهة GUI لتولد شريط الأدوات الموضح في الشكل ، إن المثال ينشئ شريط أدوات (uitoolbar) ويرسم عليه أداة التبديل (uitoggletool).

```
ht=uitoolbar(hObject);
a(:,:,1)=rand(20);
a(:,:,2)=rand(20);
a(:,:,3)=rand(20);
htt=uitoggletool(ht,'CData',a,'TooltipString','Hello');
```

في تابع الافتتاح، hobject عبارة عن وسيط الدخل الذي يحمل مقبض الشكل. إن خاصية CData تمكنك من عرض صورة بلون حقيقي على أداة التبديل.

في الإصدار الأخير لبرنامج الماتلاب MATLAB 2012b تم إضافة أداة مميزة تسمح بإضافة شريط أدوات بسهولة أكبر.



١٣,٨ برمجة القوائم

إن محرر القائمة يولد تابعاً جزئياً لاستدعاء فارغ لكل جزء قائمة ، متضمناً عناوين قائمة.

أ. برمجة عنوان القائمة

بسبب أن النقر على عنوان القائمة يعرض وبشكل أوتوماتيكي القائمة تحته، فإنك من الممكن أن لا تحتاج لبرمجة الاستدعاءات على مستوى العنوان. مع ذلك فإن الإستدعاء المرفق بعنوان قائمة يمكن أن يكون مكاناً جيداً لتمكين أو عدم تمكين أجزاء القائمة الموجودة ضمنه.

لاحظ المثال الموجود في الصورة التالية ، عندما يقوم المستخدم باختيار الخيار to file تحت القائمة edit للخيار copy. فإنه فقط الاستدعاء to file سيكون مطلوباً لإنجاز هذا الفعل.
افرض ، أنك تود أن تسمح لعناصر فقط بالنسخ إلى ملف خارجي. يمكنك أن تستخدم استدعاء Callback للجزء Copy لتمكين أو عدم تمكين الجزء to file، معتمداً على نوع العنصر المختار.

ب. فتح مربع حوار من استدعاء القائمة

إن الاستدعاء callback لجزء القائمة to file يمكن أن يتضمن الشيفرة الموضحة فيما يلي ليعرض مربع حوار معياري لتخزين الملفات :

```
[file,path]=uiputfile('animinit.m','save file name');
```

ت. تحديث اختبار جزء القائمة

إن الاختبار يكون مفيداً ليشير إلى الحالة اللحظية لبعض أجزاء القائمة. إذا اخترت check mark this item في محرر القائمة، فإن الجزء سيظهر بشكل أولي مختبراً. في كل مرة يقوم المستخدم باختيار جزء القائمة، فإن الاستدعاء لهذا الجزء يجب أن يشغل الاختبار على on أو off. يوضح المثال التالي لك كيف تفعل هذا بتغيير قيمة خاصية checked لأجزاء القائمة :

```
If strcmp(get(gcbo,'Checked'),'on')
    set(gcbo,'Checked','off');
else
    Set(gcbo,'Checked','on');
end
```

إن التابع gcbo يسترجع مقبض العنصر الذي ينفذه الاستدعاء، في هذه الحالة هو جزء القائمة. إن التابع strcmp يقارن عبارتين نصيتين و يعيد القيمة المنطقية ١ (صحيح) إذا كانتا متماثلتين و عدا ذلك يعيد ٠ (خطأ) .
إذا قمت بضبط جملة المحاور لتكون مرئية عندما يفتح المستخدم الواجهة GUI لأول مرة، تأكد من اختيار مربع الاختيار check mark this item في محرر القائمة، لذا فإن الاختبار سيظهر بعد تهيئة جزء القائمة show axes.

قيادة المعطيات و الربط بين عدة واجهات

١٤,٨ آليات قيادة المعطيات

إن أغلب الواجهات الرسومية GUI تولد أو تستخدم المعطيات الخاصة بالتطبيق. سنشرح في هذه الفقرة الآليات الثلاثة لقيادة المعطيات المعرفة للتطبيقات في بيئة GUI. إنها تؤمن وسيلة للتطبيقات من أجل تخزين و استرجاع وتبادل المعطيات المخزنة.
إن آليات معطيات الواجهة GUI و معطيات التطبيق تكون متشابهة ولكن معطيات GUI يمكن أن تكون أبسط للاستخدام. يستخدم GUIDE بشكل خاص معطيات GUI لقيادة البنية handles، و لكن تستطيع أيضاً استخدام بنية handles لمعطيات GUI أو معطيات التطبيقات لقيادة المعطيات المعرفة للتطبيق. إن خاصية UserData يمكن أن تحمل المعطيات المعرفة للتطبيق.

١,١٤,٨ معطيات الواجهة GUI

تتم قيادة معطيات الواجهة GUI باستخدام التابع guidata. حيث يستطيع هذا التابع أن يخزن متحول وحيد كمعطيات GUI. كما أنه يستخدم أيضاً لاسترجاع قيمة هذا المتحول.

١) حول معطيات الواجهة GUI تكون دائماً مرافقة لشكل الواجهة GUI، و تكون متاحة لكل استدعاءات مكونات الواجهة. إذا حددت مقبض المكونة عندما تخزن أو تسترجع معطيات الواجهة فإن MATLAB وبشكل أوتوماتيكي يرفق المعطيات مع شكل أم (parent) المكونات.
إن معطيات الواجهة تتضمن فقط متحولاً وحيداً طوال الوقت. وكتابة معطيات الواجهة يكون فوق معطيات الواجهة الموجودة. لهذا السبب فإن معطيات الواجهة تكون معرفة عادة لتكون البنية التي تستطيع بها أن تضيف حقولاً لها كيفما تريد.
إن معطيات الواجهة تزود مطوري التطبيق بالواجهة الملائمة لمعطيات تطبيق الأشكال.

٢) معطيات الواجهة GUI في GUIDE :
إن GUIDE و بشكل أوتوماتيكي يستخدم guidata ولإنشاء المحافظة على بنية handles، هذه البنية تمرر بشكل أوتوماتيكي كوسيط لكل استدعاء، حيث أن البنية handles تتضمن المقابض لكل المكونات في الواجهة GUI.
في الواجهات المبنية باستخدام GUIDE، فإنه لا يمكنك استخدام guidata لقيادة أي متحول غير البنية handles. إذا فعلت ذلك فإنه يمكن أن تكتب فوق بنية handle وبالتالي واجهتك لن تعمل. إذا كنت تريد استخدام معطيات الواجهة GUI لتشارك المعطيات المعرفة لتطبيق ضمن الاستدعاءات فإنه يجب عليك أن تخزن المعطيات في الحقول التي تريد أن تضيفها إلى بنية handles.

٣) إضافة حقول إلى بنية handles
لإضافة حقل إلى بنية handles، و التي تمرر كوسيط استدعاء في GUIDE :
- خصص قيمة للحقل الجديد المضاف إلى البنية. على سبيل المثال :

```
handles.number_errors=0;
```

أضف الحقل number_errors إلى البنية handles وضع فيه قيمة مساوية للصفر.

- استخدم الأمر التالي لتخزين المعطيات.

```
guidata(hObject,handles)
```

عندما يكون hObject مقبض المكونة التي شغلها الاستدعاء. فإنه سيتم تمريره بشكل أوتوماتيكي لكل استدعاء.

٤) تغيير معطيات الواجهة GUI في ملف M-file المولد من GUIDE
في GUIDE المولد لملف M-file، فإن معطيات الواجهة GUI تمثل بشكل دائم بالبنية handles. هذا المثال يحدث البنية handles ومن ثم يخزنها :

- افرض أن البنية handles تتضمن الحقل المعرف للتطبيق handles.when الذي يملك القيمة 'now'.
- في استدعاء الواجهة GUI، نفذ التغيير المطلوب handles. هذا المثال يغير قيمة handles.when إلى 'later'، ولكنه لا يقوم بتخزين البنية handles.

```
handles.when='later';
```

- خزن النسخة المغيرة للبنية handles بالأمر :

```
Guidata(hObject,handles)
```

حيث أن hObject الذي يمرر بشكل أوتوماتيكي إلى كل استدعاء، يكون المقبض للمكونة التي شغلها الاستدعاء. إذا لم تخزن البنية handles ، فإن التغيرات التي نفذتها في الخطوة السابقة ستضيع.

١٥,٨ صيغة الربط بين عدة واجهات Multipages GUI forms

من خلال المثال التالي سنتعلم كيف يمكن إنشاء GUI لصفحات/شرائح مختلفة ، وأيضاً إنشاء واجهة رئيسية master gui لتتحكم بعدة واجهات .

- handles

لكل واجهة رسومية GUI بنية (structure) تدعى : handles ، و لكي نتمكن من رؤية أو تحرير هذه البنية يمكن استخدام **guidata**

☒ مقبض الحصول على الخاصية GET HANDLES

```
h=guidata (gui_reference);
```

☒ مقبض ضبط الخاصية SET HANDLES

```
guidata (gui_reference,h);
```

حيث gui_reference هو رقم مزدوج يعمل كمؤشر يشير إلى gui ، لتقوم بتخزين هذا المؤشر كمتغير يقوم باستدعاء الـ gui بواسطة الـ output .

```
gui_reference=gui1;
```

- Creating Slide Pages

- قم بإنشاء واجهتين 2guis (GUI1.m and GUI2.m) .
- وضع في كل منهما الكائنات التالية:

- two edit box : edit1 , edit2.
- two pushbutton : pushbutton1 with title Next , pushbutton2 with title Previous.

- الآن من أجل عملية التصفح سنضيف في كل gui :

next and prev

- قم بإضافة السطرين التاليين في التابع OpeningFcn لكل من الواجهتين اللتين قمت بإنشائهما gui1 ,gui2

```
function gui1_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to gui2 (see VARARGIN)

% Choose default command line output for gui2
handles.output = hObject;
handles.next=1;
handles.prev=1;
% Update handles structure
guidata(hObject, handles);
```

- الآن اذهب إلى pushbutton1_callback وأضف هذه الأسطر لعرض الشريحة التالية وإخفاء نفسه :

```
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
set(handles.next,'visible','on');
set(handles.output,'visible','off');
```

- ومن أجل pushbutton2_callback تماماً كالسابق لكن لعرض الشريحة السابقة :

```
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
set(handles.prev,'visible','on');
set(handles.output,'visible','off');
```

- Creating Master Slide

- قم بإنشاء gui خالية أضف إليها pushbutton بعنوان start واحفظها (guio.fig) ، وبشكل آلي سيتم إنشاء الملف guio.m .
- الآن في هذا الملف ، اذهب إلى guiOpeningFcn ، وستجد الكود التالي :

```
function gui0_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to untitled (see VARARGIN)

% Choose default command line output for untitled
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
```

- قم بتحرير هذا الكود ،لاستدعاء الشريحتين وحفظ مؤشرات تشير لهم فيه .

```
% Choose default command line output for guio
handles.output = hObject;

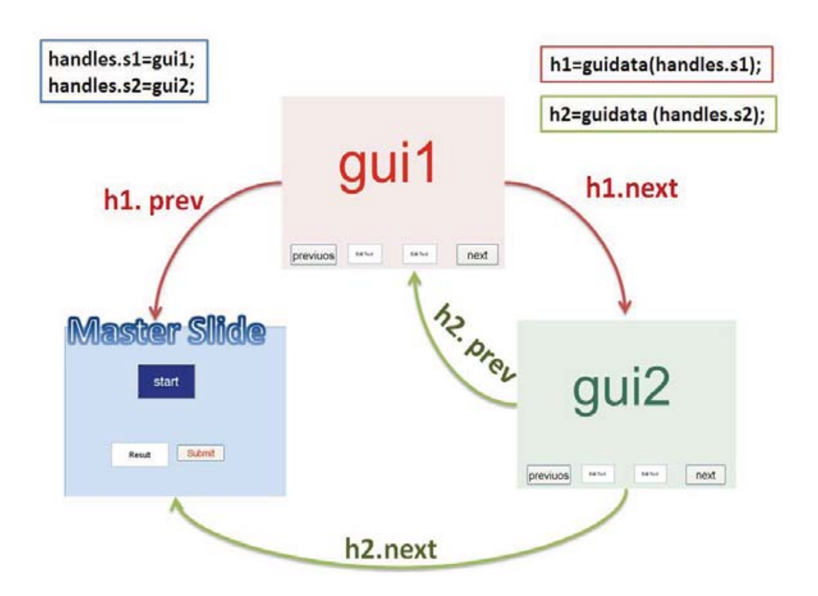
handles.s1=gui1;
handles.s2=gui2;

h1=guidata(handles.s1);
h1.next=handles.s2;
h1.prev=hObject;
guidata(handles.s1,h1);

h2=guidata(handles.s2);
h2.next=hObject;
h2.prev=handles.s1;
guidata(handles.s2,h2);

% Update handles structure
guidata(hObject, handles);
handles.output;
%set(handles.output,'Visible','off');
set(handles.s1,'visible','off');
set(handles.s2,'visible','off');
guidata(hObject,handles);
```

- كما ترى ، لقد استدعيت gui1 و gui2 وحفظت مصدرهم أو مؤشرات تعود إلى handles.s1 و handles.s2 .
- ومن أجل التصفح .. أشرنا إلى s2 كشريحة تالية للشريحة s1(slide 1) والشريحة الرئيسية mastergui كشريحة سابقة لـ s1، و بنفس الطريقة لـ s2(slide 2) .



- الآن نحتاج للتعديل على كود زر البداية start button لإخفاء الصفحة الرئيسية ونبدأ slide 1 .
هنا ببساطة أضف إلى pushbutton1_Callback الموجود في gui0.m الأسطر التالية :

```
set(handles.output,'visible','off');
set(handles.s1,'visible','on');
set(handles.s2,'visible','off');
```

- تذكر أن : handles.output يُشير إلى مؤشر gui نفسه .

- Final Touch: Processing your data

- بينما أنت تعمل .. ستعود إلى الصفحة الرئيسية مرة أخرى لأن الشريحة التالية لـ s2 هي masterpage الصفحة الرئيسية ، هنا تحتاج لجمع كل البيانات ولتتمكن من استخدامها كما تريد .
- على سبيل المثال .. أريد أن أضيف الأرقام الأربعة المكتوبة في 4 editboxes إلى بعضها ، أي جمع الأرقام المكتوبة في gui1 و gui2 الموجودة ضمن الـ editboxes .
- للسهولة ، ضع زر آخر pushbutton باسم Submit ، وأيضاً Editbox بعنوان Result ، على الصفحة الرئيسية masterpage ، واكتب كود الحساب في Submit_callback .
- لا حاجة لأن تقول أنك تحتاج لضغط هذا الزر submit_button عندما تعود إلى الصفحة الرئيسية لتعمل التحرير النهائي .
- هذا كود الحساب :

```
h1=guidata(handles.s1);
a1=get(h1.edit1,'string');
a2=get(h1.edit2,'string');
h2=guidata(handles.s2);
a3=get(h2.edit1,'string');
a4=get(h2.edit2,'string');
a=str2num(a1)+str2num(a2)+str2num(a3)+str2num(a4);
set(handles.result,'string',a);
```

- فائدة الصفحة الرئيسية أنها تتيح لك أن تعيد ترتيب وتنظيم شرائحك بسهولة كبيرة ،وكل gui بشكل مستقل .



التحكم بالمنافذ و الربط مع المتحكمات

- الربط مع النافذة التفرعية .
- الربط مع النافذة التسلسلية .
- الربط مع المتحكمات الصغرية .
- أداة التطوير tmtool.
- الأردوينو Arduino.

التحكم بالمنفذ التفرعي

Controlling with Parallel Port

١,١٠ مقدمة

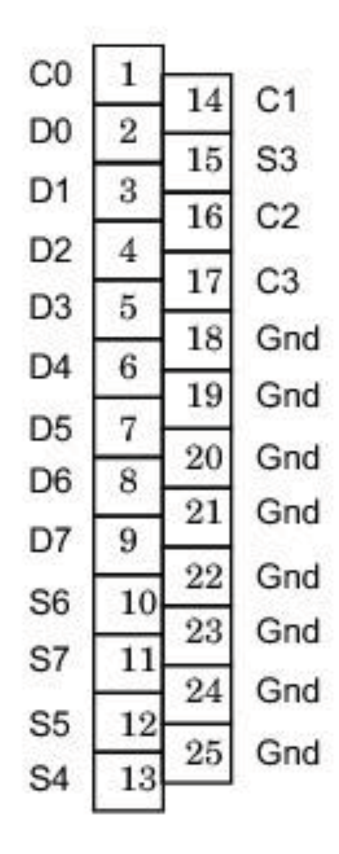
يعد المنفذ التفرعي هو الأكثر شيوعاً لسهولة استخدامه والتحكم به في المشاريع الصغيرة التي قد تُنفذ في المنازل ، حيث يسمح لنا هذا المنفذ بإدخال بيانات بحزمة بعرض ٩ بت أو بإخراج بحزمة بعرض ١٢ بت في أي لحظة ، كما يحوي هذا المنفذ على أربع خطوط تحكم وخمس خطوط حالة و ثمان خطوط لتبادل المعطيات ، وفي أغلب الحواسيب نشاهد هذا المنفذ في لوحة المنافذ الخلفية والنوع الأكثر انتشاراً هو D-type الذي يحوي 25pin (male or Female).
وقد يوظف هذا المنفذ في كثير من الأمور مثل التحكم بدارات إلكترونية أو التحكم بالأنواع المختلفة للمحركات و النظم غير المعقدة ، كما يستخدم في التحكم بمبرمجات PIC/Atmel وفي الأتمتة.

٢,١٠ المنفذ التفرعي

تعتبر بوابة الطابعة بوابة للخرج فقط ولكن لحسن الحظ بالإضافة إلى البتات الثمانية المتوفرة كمخارج رقمية فإنها تمتلك العديد من خطوط المصافحة و التي لا يقل عددها عن تسعة خطوط خمسة مداخل وأربعة مخارج و الشكل الجدول يوضح توزع الخطوط في موصل الطابعة المؤلف من ٢٥ خط من النوع D.

Hardware Inverted	Register	Direction In/out	Pin No (D-Type 25)
Yes	Control	In/Out	1
	Data	Out	2
	Data	Out	3
	Data	Out	4
	Data	Out	5
	Data	Out	6
	Data	Out	7
	Data	Out	8
	Data	Out	9
	Status	In	10

Yes	Status	In	11
	Status	In	12
	Status	In	13
Yes	Control	In/Out	14
	Status	In	15
	Control	In/Out	16
Yes	Control	In/Out	17
		Gnd	18 – 25



D : Data (in,out) - C : Control - S : Status - Gnd : Ground

وبهذه الطريقة يمكننا التمييز بين أرجل المعطيات و أرجل التحكم و الأرجل المؤرضة.

٣,١٠ التحكم بالمنفذ التفرعي بواسطة الماتلاب

- يحتاج الماتلاب لتعريف المنفذ الذي ستتعامل معه التعليمة التالية :

```
>> dio=digitalio('parallel',1)
```

إن التعليمة digitalio تعرف المنفذ الرقمي فنكتب اسم المنفذ وهو هنا المنفذ التفرعي ثم نكتب رقمه أي LPT1 أو LPT2 وهنا تم تعريف LPT1.

- بعد تعريف المنفذ علينا تحديد الأرجل التي نحتاجها وتعريفها هل هي دخل أم خرج ويمكننا القيام بذلك من خلال التعليمة التالية :

```
>> addline(dio,[0:7],'out')
```

حيث أن هذه التعليمة قامت بجعل الأرجل من ٢ إلى ٩ كأرجل خرج وهي ترقم في الماتلاب من ٠ إلى ٧.

- في الخطوة الأخيرة قمنا بتحديد أرجل الخرج والآن ولو أردنا أن نخرج قيمة معينة (0 or 1) من هذه الأرجل فما علينا إلا ان نستعمل التعليمة التالية :

```
>> putvalue(dio,[1 1 1 1 0 0 0 0])
```

بهذه الطريقة نكون قد أخرجنا القيمة التي نريد من المنفذ التفرعي. إذا أردنا الآن أن نقرأ معطيات من جهاز ما فيمكننا ذلك بالتعليمة التالية وذلك بعد تحديد الأرجل التي نريد القراءة منها :

```
>> getvalue(dio)
```

- برنامج الإخراج :

إن كتابة ثلاث تعليمات عند كل إخراج وإدخال أمر ممل ولذلك يجب تجهيز برنامج للإخراج وآخر للإدخال وسنكتب برنامج كامل لإخراج المعطيات في m-file :

```
function c=out(a,n)
dio=digitalio('parallel',1);
addline(dio,a,'out');
putvalue(dio,n);
```

- a : أرقام الأرجل التي تريد التخريج عليها وتدخل a كمصفوفة تحوي أرقام الأرجل.
- n : البيانات التي تريد إخراجها ويمكنك إدخالها بالشكل العشري أو الثنائي.

- برنامج الإدخال :

البرنامج التالي هو لقراءة المعطيات المدخلة :

```
function c=in(a)
dio=digitalio('parallel',1);
addline(dio,a,'in');
```

```
getvalue(dio);
```

٤,١٠ امثلة عملية

- مثال (١)

برنامج للتحكم بإضاءة ثمان ليدات موصولة إلى الأرجل (٢ – ٩) ويقوم بإضاءة الليد الأول ثم يطفئه ويضيء الليد الثاني و هكذا إلى الثامن ...

```
funcion flash2(a)
for i=1:a
   for t=0:7
      c=2^t
      out([0:7],c)
      pause(0.05)
   end
end
out([0:7],0)
```

في هذا البرنامج استدعينا برنامج out الذي كتبناه سابقاً وحفظناه وكان الاستدعاء هو فقط ذكر اسم البرنامج مع عناصر الدخل أو المتغيرات.

a : عدد مرات تكرار البرنامج.   استخدمنا المتغير t لكي لا نكتب الأعداد الثنائية في كل مرة.

قيمة t	قيمة c	التحويل الثنائي للمتغير c
0	1	00000001
1	2	00000010
2	4	00000100
3	8	00001000
4	16	00010000
5	32	00100000
6	64	01000000
7	128	10000000

إن التعليمة pause هي تعليمة تأخير زمني واحدته الثانية فتحدد له كم ثانية تريد التأخير حتى تلحظ التغييرات على الليدات.

و التعليمة الأخيرة في البرنامج هي من أجل إطفاء جميع الليدات بعد إنهاء البرنامج.

بطريقة ثانية :

يمكن كتابة البرنامج بطريقة أخرى كالتالي :

```
function flash12(a)
```

```
for i=1:a
   for t=1:8
      h=zeros(1,8);
      h(t)=1;
      out([0:7],h)
      pause(0.05)
   end
end
out([0:7],0)
```

في هذا البرنامج تم استخدام مصفوفة صفرية من ثمان عناصر في كل مرة يتم تحويل أحد هذه الأصفار ليصبح واحداً وهو يعبر عن رقم الليد الواجب إضاءته.

- مثال (٢)

برنامج بتحويل الأعداد العشرية المتزايدة إلى الشكل الثنائي عن طريق إضاءة الليدات ...

```
function flash1(a)
for i=1:a
   for t=1:255
      out([0:7],t)
      pause(0.05)
   end
end
out([0:7],0)
```

إن الإدخال هنا تم بشكل عشري.

- مثال (٣)

برنامج يقوم بإضاءة الليدات من الطرفين إلى المنتصف ...

```
function flash(a)
for i=1:a
   out([0:7],255)
   pause(0.25)
   out([0:7],[0 1 0 0 0 0 1 0])
   pause(0.25)
   out([0:7],[0 0 1 0 0 1 0 0])
   pause(0.25)
   out([0:7],[0 0 0 1 1 0 0 0])
   pause(0.25)
end
out([0:7],0)
```

التحكم بالمنفذ التسلسلي

Controlling with Serial Port

٥,١٠ مقدمة

الاتصالات التفرعية

سرعة النقل عالية جداً ولكن مسافة نقل البيانات تكون صغيرة جداً ومحدودة (بازدياد طول الناقل تتشكل سعات وضجيج عالي على المسارات ، كما أن حجم الناقل يصبح كبير والكلفة عالية وبالنتيجة تتشوه الإشارة).

الاتصالات التسلسلية

تستخدم الاتصالات التسلسلية على نطاق اوسع بكثير (من التفرعية) وتمتاز بمناعة عالية ضد الضجيج ونقل البيانات لمسافات كبيرة وحجم الناقل صغير وبالتالي الكلفة ضئيلة.

٦,١٠ مفاهيم أساسية

- الاتصالات المتواقتة :

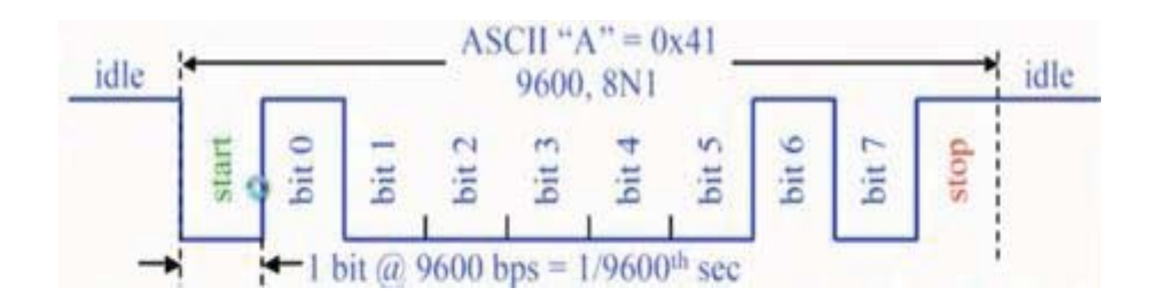
١- تحوي خطين على الأقل (خط إضافي دوماً للتزامن).

٢- سرعة النقل لها علاقة بالتردد.

- الاتصالات اللامتواقتة :

١- لا تحوي خط تواقت إضافي.

٢- يتم التزامن بها من خلال معدل نقل متعارف عليه بين الطرفيتين يدعى Frame



- Half-Duplex الاتصال أحادي الاتجاه
- Full-Duplex الاتصال ثنائي الاتجاه

- خانة الإيجابية Parity Bit : تستخدم هذه الخانة للفحص والتحقق ويمكن أن تكتشف الأخطاء بنسبة ٥٠%. وهي خانة يضيفها المرسل ويستخدمها المستقبل لضمان عدم ضياع المعلومات وتتعلق خانة الإيجابية بعدد الواحدات في البايت المرسل ولها حالتين :

- زوجي Evan : تملك الخانة القيمة صفر إذا كان عدد الواحدات في البايت المرسل زوجي وإلا فقيمتها ١.
- فردي Odd : تملك الخانة القيمة صفر إذا كان عدد الواحدات في البايت المرسل فردي وإلا فقيمتها ١.

- عدد البتات لكل محرف (N) :

ليس بالضرورة (8 bits) وحدوده :

9 bits → 5 bits ، 9bit وجود بت إشارة إضافي

- خانة بت التوقف Stop Bit

يُعلم المرسل من خلالها المستقبل بانتهاء عملية الإرسال ويمكن أن و يمكن أن يكون هذا البت (1 bit or 1.5 bits or 2 bits).

- سرعة النقل Baud Rate

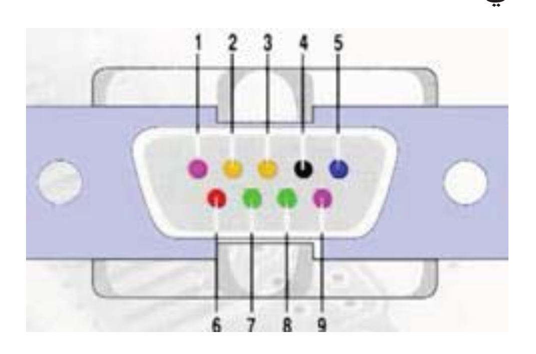
عدد البتات المرسلة خلال ثانية واحدة على خط اتصال تسلسلي

300 → 115200 bps

الزمن اللازم لإرسال بت واحد $Bit_{Time}=\frac{1}{Baud\ Rate}$

عدد البايتات بالثانية $Byte=\frac{Baud\ Rate}{8}$

- مخطط المنفذ التسلسلي :



(المنفذ التسلسلي COM – وظائف الاقطاب)

Pin	Name	Direction	Function	Description
1	CD	In	Control	Carrier Detect
2	RXD	In	Data	Receive Data
3	TXD	Out	Data	Transmit Data
4	DTR	Out	Control	Data Terminal Ready
5	GND	---	Ground	System Ground
6	DSR	In	Control	Data Set Ready
7	RTS	Out	Control	Request to Send
8	CTS	In	Control	Clear to Send
9	RI	In	Control	Ring Indicator

يعتمد المنفذ التسلسلي على بروتوكول الاتصال RS-232 وهذا البروتوكول له الميزات :

١- بروتوكول اتصال تسلسلي غير متزامن.

٢- يستخدم ترميز ASCII (أي يجب استقبال محرف).

٣- يعمل فقط عند وجود معطيات (توفير طاقة).

٤- يستخدم ثلاث خطوط فقط وهم :

Rx : مستقبل

Cx : مرسل

GRD : أرضي

٥- مستويات الجهد :

(0) → +3v ~ +25v

(1) → -3v ~ -25v

+3v ~ -3v → مستوى غير معروف

وبالتالي نحتاج محول لنستطيع التعامل مع دارات TTL.

٦- توتر اللاحمل المسموح لا يتجاوز ±٢٥فولط وتيار القصر ٥٠٠ميلي أمبير وعدا ذلك يعطل المنفذ.

<u>**وبالنتيجة ولكل نستطيع الإرسال والاستقبال عبر المنفذ التسلسلي يجب تحديد :**</u>

١- نمط الإرسال (باتجاه واحد – أو باتجاهين).

٢- عدد البتات (٦بت – ٧بت – ٨بت – ٩بت).

٣- سرعة الإرسال.

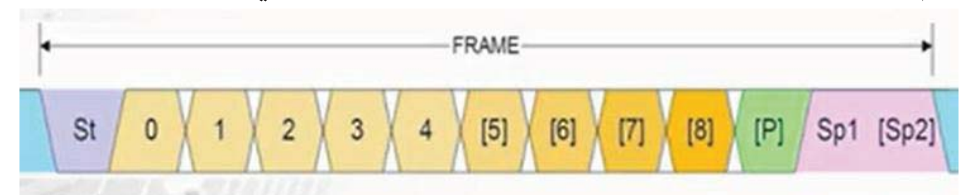
٤- فحص الإيجابية Parity Bit

٥- عدد بتات التوقف Stop Bit

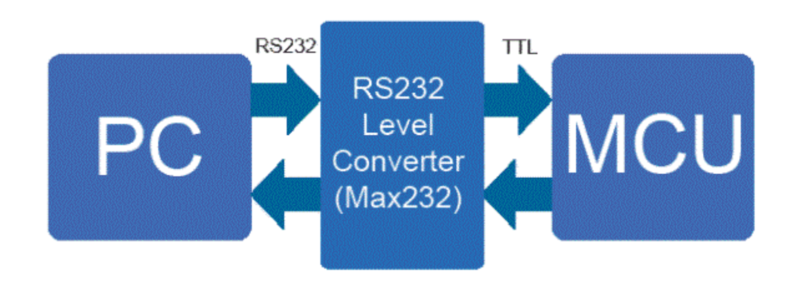
٧,١٠ النافذة اللاتزامنية من طرف المتحكمات UART

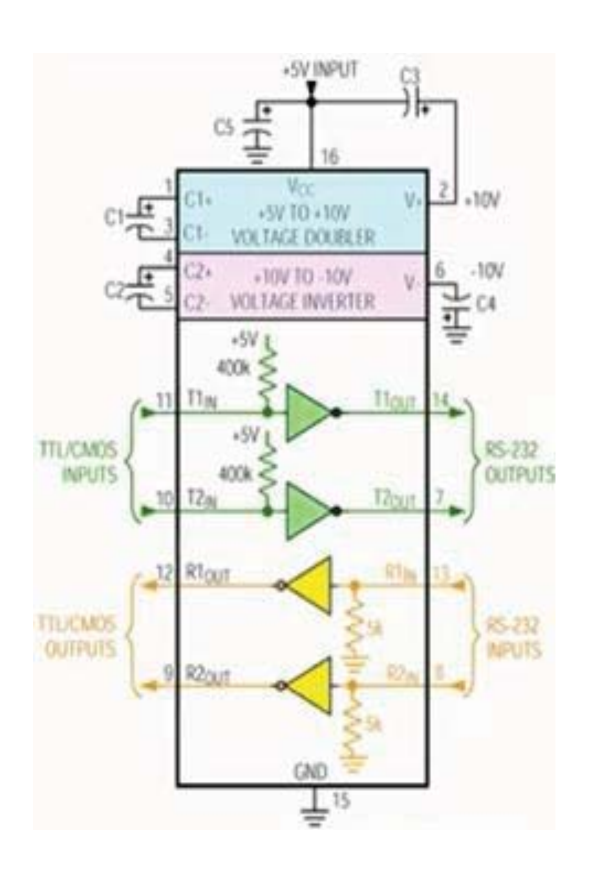
UART : Universal Asynchronous Receiver and Transmitter Interface.

تعتبر هذه النافذة من أكثر نوافذ الاتصال التسلسلي استخداماً في الأنظمة الرقمية ومبدأ عملها وكذلك بروتوكولها توافق تماماً مع RS-232 إلا أن المستويات المنطقية فيها وفق منطق TTL ولذلك تستخدم دارات التحويل والملائمة كوسيط بين المنفذ التسلسلي وبين النافذة التسلسلية.



تستخدم الدارة المتكاملة MAX232 من أجل الملائمة بين الطرفين أي التحويل بين RS232 <> TTL.





٨,١٠ التحكم بالمنفذ التسلسلي من البيئة البرمجية في الماتلاب

- Serial

- هذا التابع هو البنية الأساسية للاتصال .
- يعمل دوماً دون أخطاء ، حتى في حال عدم وجود المنفذ المطلوب.
- يعيد هذا التابع كافة الخصائص و هي:

```
Serial Port Object : Serial-COM6
Communication Settings
Port:            COM6
BaudRate:        9600
Terminator:      'LF'
Communication State
Status:          closed
RecordStatus:    off
Read/Write State
TransferStatus:  idle
BytesAvailable:  0
ValuesReceived:  0
ValuesSent:      0
```

- لاحظ من الخصائص أن النافذة التسلسلية مغلقة.

هذا التابع مسؤول عن إنشاء منفذ تسلسلي تسلسلي في الماتلاب ، حيث ينشأ هذا التابع منفذ تسلسلي يتعامل مع المنفذ المحدد وفي حال كان المنفذ غير موجود أو كان مستخدماً من قبل جهاز آخر فلن يتمكن برنامج الماتلاب من الاتصال معه باستخدام تابع الاتصال.

```
Obj=serial('port')
Obj=serial('port','propertyName',PropertyValue,…)
```

حيث :

اسم المنفذ التسلسلي : 'Port'

خصائص اسم المنفذ التسلسلي : propertyName

قيمة الخاصية : PropertyValue

و يمكن التعديل في الخصائص كما يلي :

```
>> set(s,'baudrate',4800);
>> s.baudrate=4800;
```

يمكننا مباشرة التحكم بالخصائص عند إنشاء النافذة كما يلي : -

```
>> s=serial('com6','baudrate',4800);
```

```
>> get(s,'baudrate');
 ans = 4800
>> s.baudrate
 ans = 4800
```

مثال :

```
s=serial('com1')
get(s,{'Type','Name','Port'})
      ans =
              'serial'    'serial-com1'    'com1'
```

- fopen

فتح المنفذ التسلسلي من برنامج الماتلاب وجعله متصلاً وجاهز للإرسال والاستقبال.
لتغيير حالة النافذة من الوضعية المغلقة للوضعية المفتوحة نستخدم التابع :

```
>> fopen(s)
```

يمكننا تفحص خصائص الحالة للنافذة التسلسلية من خلال كتابة :

```
>> s.Statues
      ans = open
```

fopen(obj)

جهاز التوصيل obj : قبل ان نتمكن من القراءة والكتابة على المنفذ يجب الوصل مع الآلة باستخدام هذا التابع ، حيث عند التوصيل obj :
١- المعلومات تبقى في ذاكرة الدخل أو الخرج نشطة.
٢- الحالة (statues) تضبط على أنها مفتوحة وجاهزة.
٣- الخصائص :
(ByeAvailable – ValueReceived – ValueSent and Bytestooutpu)
تأخذ القيمة (0)
هناك بعض الخصائص التي يجب الانتباه أنها للقراءة فقط كما أن و يجب تحديدها قبل فتح الاتصال بالتابع fopen وهناك خصائص اخرى يجب تحديدها بعد فتح الاتصال.

- fclose

إغلاق الاتصال مع الآلة (إغلاق المنفذ).

Fclose(obj)

جهاز التوصيل obj :
في حال تم قطع الاتصال بنجاح فإن خاصية الحالة ستغلق ويمكن إعادة الاتصال من جديد بتابع الاتصال.

- إرسال الإشارات fwrite , fprintf

يمكن من خلال برنامج الماتلاب كتابة أي نوع من المعطيات على المنفذ التساسلي سواءً القيم الثنائية ، المحرفية ، الصحيحة ، و العائمة و بدقة محددة ، ويستخدم التابعين fwrite & fprintf من أجل ذلك.

لكتابة قيم صحيحة أو عائمة نستخدم التابع :

```
>> fwrite(s,vector_array,'precision');
```

- الـ precision تحدد لتكون :

int8 – int16 – float32 – float64 – uint8 – char

- لكتابة قيم محرفية (نصية) نستخدم التابع :

```
>> fwrite(s,'string');
```

لكتابة المحارف كما يلي : fprintf ويمكن استخدام التابع

```
>> fprintf(s,'string');
```

- ملاحظة خاصة ببرمجة الأردوينو :

استخدم التابع println و لا تستخدم التابع print من أجل الإرسال التسلسلي عبر النافذة التسلسلية من الأردوينو للماتلاب.

- إستقبال الإشارات fscanf ,fread

- يستخدم التابع fscanf لقراءة المعطيات :

```
>> fscanf(s);
```

و يقرأ هذا التابع كافة البيانات لحين الوصول لسطر جديد.

- ولذلك يتم استخدام التابع println في برمجة الأردينو بدلاً من التابع print من أجل الإرسال التسلسلي عبر النافذة التسلسلية من الأردينو للماتلاب.
- وبهذه الطريقة سيقوم الماتلاب تلقائياً لتحويل البيانات لأفضل صيغة و تخزينها كمتغيرات في الماتلاب.
- وعندما لا يكون هناك معلومات تقرأ (تصل للماتلاب عبر النافذة التسلسلية) ، سيحدث تعليق للبرنامج.
- يمكن منع حدوث تعليق للبرنامج من خلال تفحص خاصية byteAvailable للمنفذ التسلسلي بشكل لحظي.

```
If ByteAvailable > 0
Data=fscanf(s);
end
```

- يمكن أيضاً استخدام التابع fread بدلاً من fscanf ولكن في هذه الحالة لا يتم تحويل البيانات لأفضل صيغة بشكل تلقائي و يجب القيام بذلك يدوياً بتحديد ذلك كما يلي:

```
>> fread(s,size,'precision')
```

- Delete & fclose

- يستخدم التابع fclose من أجل إغلاق النافذة التسلسلية.

```
>> fclose(s)
```

- حذف الكائن الخاص بالمنفذ التسلسلي من خلال التابع :

```
>> delete(s)
```

يجب دوماً إغلاق المنفذ بعد فتحه و إنهاء تبادل المعطيات و إلا فإننا سنواجه مشاكل مع النافذة التسلسلية عند محاولة الاتصال مجدداً.

٩,١٠ حماية البيانات من الضياع :

- حدد بدقة حجم الذاكرة (buffer) اللازمة ، حيث كما ذكرنا يتم حذف أقدم متغير بالذاكرة عن أمتلائها و محاولة الكتابة بها مجدداً.

```
Set(s,'BufferSize',1024)
s.BufferSize=1024;
```

- يجب ان يتساوا معدل نقل البيانات في الماتلاب مع معدل نقل البيانات من المتحكم (الأردوينو مثلاً) (Baudrate).
- يتم إرسال و استقبال البيانات عبر النافذة التسلسلية مع الماتلاب بشكل دوري و يمكن تفحص حالة البيانات المرسلة و كذلك المستقبلة من خلال التابعين :

ValuesSent & ValuesReceived

- يمكن التأكد كذلك من اكتمال تبادل البيانات من خلال الخاصية :

TransferStatus

- التابع fprintf يمنع تنفيذ أي سطر برمجي عند تبادل البيانات في حين لا يقوم التابع fwrite بذلك.

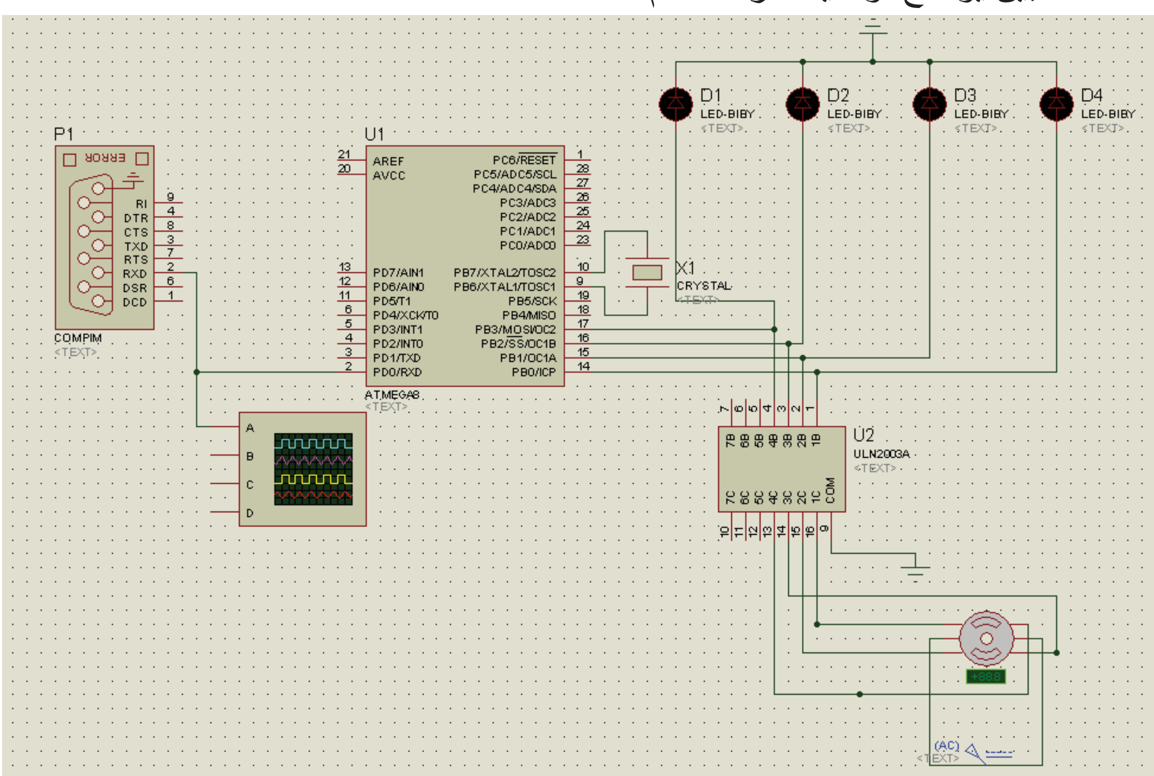
مثال (١) :

سنقوم فيها يلي بتجهيز بروتوكول اتصال بسيط للربط بين الحاسب و المتحكم Atmega8 و ذلك للتحكم بمحرك خطوي ، حيث يستقبل المتحكم الإشارة (محارف) من البيئة البرمجية في الماتلاب عبر النافذة التسلسلية و بناءً على بروتوكول الاتصال المتفق عليه يتم إصدار أوامر الإخراج.

سنبين :

- الكود البرمجي الخاص بالمتحكم
- الكود البرمجي للتحكم بالمنفذ من الماتلاب

الشكل المبين يوضح توصيل دارة التحكم.



برنامج المتحكم بواسطة المترجم Codevision كالتالي (يتم التعديل في الجزء بالمقاطعة الخاصة بالاتصال اللامتزامن كما هو موضح) :

```
// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
char status,data;
status=UCSRA;
data=UDR;
while(1)
{
if (data=='a')
{
    PORTB =0b1000;
    delay_ms(100);
    PORTB =0b0100;
    delay_ms(100);
    PORTB=0b0010;
```

```
        delay_ms(100);
        PORTB=0b0001;
        delay_ms(100);
    }
    if (data=='b')
    {
        PORTB =0b1000;
        delay_ms(200);
        PORTB=0b0100;
        delay_ms(200);
        PORTB =0b0010;
        delay_ms(200);
        PORTB=0b0001;
        delay_ms(200);
    }
    if (data=='c')
    {
        PORTB =0b1000;
        delay_ms(300);
        PORTB=0b0100;
        delay_ms(300);
        PORTB =0b0010;
        delay_ms(300);
        PORTB=0b0001;
        delay_ms(300);
    }
    if (data=='d')
    {
        PORTB =0b0001;
        delay_ms(100);
        PORTB =0b0010;
        delay_ms(100);
        PORTB=0b0100;
```

```
    delay_ms(100);
    PORTB=0b1000;
    delay_ms(100);
}
if (data=='e')
{
    PORTB =0b0001;
    delay_ms(200);
    PORTB=0b0010;
    delay_ms(200);
    PORTB =0b0010;
    delay_ms(200);
    PORTB=0b1000;
    delay_ms(200);
}
if (data=='f')
{
    PORTB =0b0001;
    delay_ms(300);
    PORTB=0b0010;
    delay_ms(300);
    PORTB =0b0100;
    delay_ms(300);
    PORTB=0b1000;
    delay_ms(300);
}
};
```

برنامج الماتلاب وتبعاً لبروتوكول الاتصال المتفق عليه يجب أن يرسل أي المحارف a,b,c,d,e or f وبناءً على ذلك نكتب البرنامج التالي :

```
s=menu('choose your state :' , '1' , '2' , '3' , '4' , '5' , '6')
s2 = serial('COM1', 'BaudRate', 9600);
fopen(s2);
if s==1
fprintf(s2,'%s','a');
```

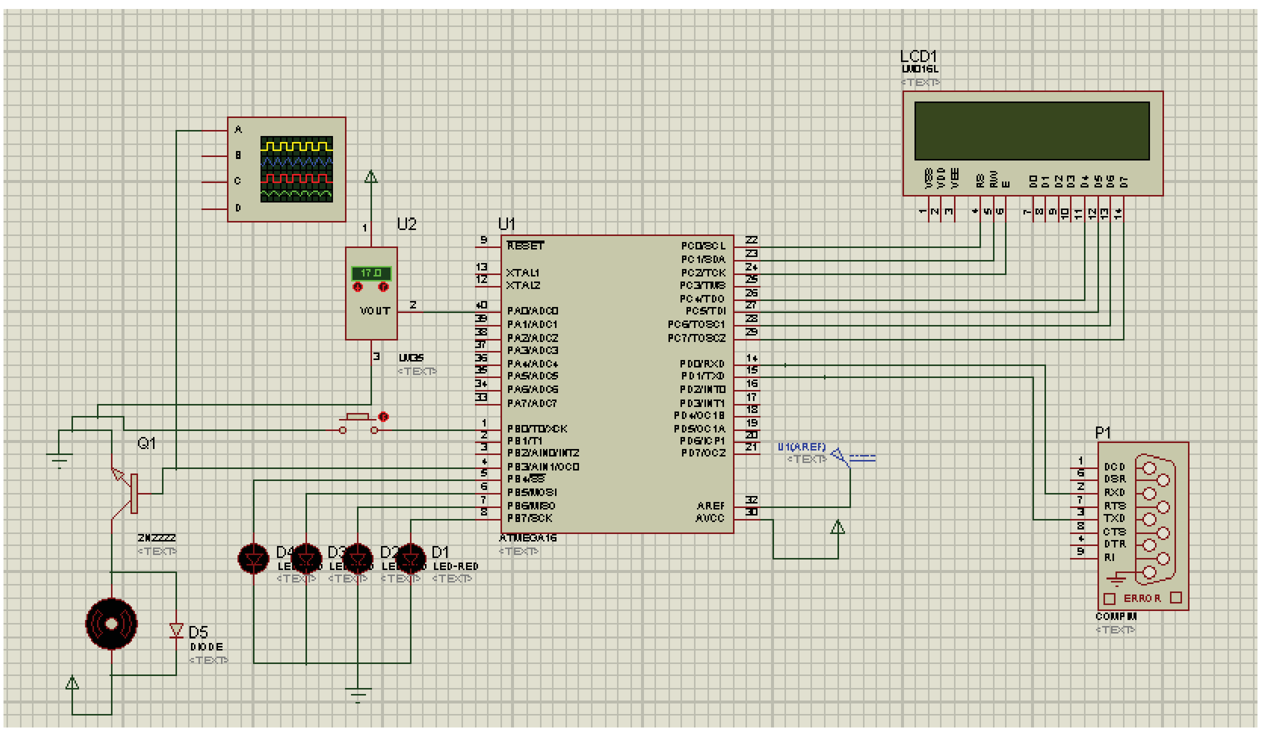
```
elseif s==2
    fprintf(s2,'%s','b');
elseif s==3
    fprintf(s2,'%s','c');
elseif s==4
    fprintf(s2,'%s','d');
elseif s==5
    fprintf(s2,'%s','e');
elseif s==6
    fprintf(s2,'%s','f');
end
fclose(s2);
delete(s2);
```

مثال (٢) :

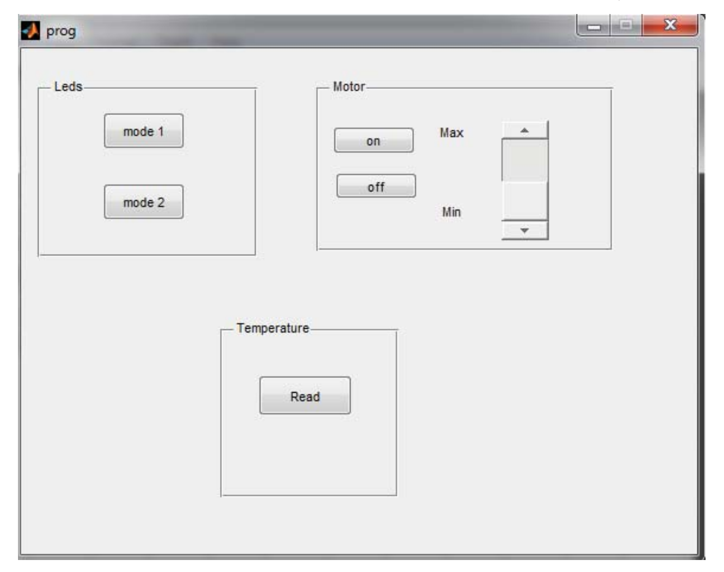
التحكم بهذه الدارة من خلال واجهة رسومية تم انشائها باستخدام الماتلاب حيث نتحكم بنمط إضاءة الليدات حيث يوجد نمطين ، ونتحكم بسرعة المحرك من خلال المنزلق في الواجهة الرسومية أما فيزيائياً فالتحكم بسرعة المحرك يتم من خلال نبضات PWM ، ونقرأ قيمة حساس الحرارة ونظهرها على شاشة LCD وكذلك على واجهة الماتلاب .

ربط هذه الدارة مع الحاسب تم من خلال المنفذ RS-232 ومحاكاة ذلك برمجياً تتم من خلال برنامج virtual serial port الذي يقوم بإنشاء منفذين افتراضيين COM 1 , COM 2 ربطنا الماتلاب مع أحدهما والبروتوس مع الآخر.

الكود الخاص بالمتحكم تم كتابته باستخدام Code vision والشكل التالي يوضح الدارة :



الواجهة الرسومية "الماتلاب" :



برنامج المتحكم بواسطة المترجم Codevision كالتالي :

```
#include <mega16.h<
#include <delay.h<
 //Alphanumeric LCD Module functions
#include <alcd.h<
void main(void(
}char x=0,y=0؛
 //Declare your local variables here
PORTA=0x00؛
DDRA=0x00 ؛
PORTB=0x01؛
DDRB=0xF8
PORTC=0x00؛
DDRC=0x00؛
 PORTD=0x00؛
DDRD=0x00؛
TCCR0=0x6D؛
TCNT0=0x00؛
OCR0=0x00؛
TCCR1A=0x00؛
TCCR1B=0x00؛
TCNT1H=0x00؛
```

```
TCNT1L=0x00؛
ICR1H=0x00؛
ICR1L=0x00؛
OCR1AH=0x00؛
OCR1AL=0x00؛
OCR1BH=0x00؛
OCR1BL=0x00؛
ASSR=0x00؛
TCCR2=0x00؛
TCNT2=0x00؛
OCR2=0x00؛
MCUCR=0x00؛
MCUCSR=0x00؛
TIMSK=0x03؛
UCSRA=0x00؛
UCSRB=0xD8؛
UCSRC=0x86؛
UBRRH=0x00؛
UBRRL=0x33؛
ACSR=0x80؛
SFIOR=0x00؛
ADMUX=ADC_VREF_TYPE & 0xff؛
ADCSRA=0x87؛
SPCR=0x00؛
TWCR=0x00؛
lcd_init(16؛(
 //Global enable interrupts
#asm("sei("
while (1(
}
 //       Place your code he
            if(PINB.0==0(
}            while(PINB.0==0؛(
```

```
        y=read_adc(0؛(
        lcd_putchar(y؛(
{
        if(rx_counter!=0(
        x=getchar؛()
        if(x=='g('
}
        lcd_clear؛()
        lcd_puts("Mode 1؛  ("
        PORTB.7=0؛
        PORTB.4=1؛
        delay_ms(200؛(
        PORTB.4=0؛
        PORTB.5=1؛
        delay_ms(200؛(
        PORTB.5=0؛
        PORTB.6=1؛
        delay_ms(200؛(
        PORTB.6=0؛
        PORTB.7=1؛
        delay_ms(200؛(
{
        else if(x=='h('
}
        lcd_clear؛()
        lcd_puts("Mode 2؛("
        PORTB.4=1؛
        PORTB.5=1؛
        PORTB.6=0؛
        PORTB.7=0؛
        delay_ms(300؛(
        PORTB.4=0؛
        PORTB.5=0؛
```

```
        PORTB.6=1؛
        PORTB.7=1؛
        delay_ms(300        ؛(
{
        else if(x=='m('
}
        while(1(
}
        lcd_clear؛()
        lcd_puts("Motor؛("
        delay_ms(10؛(
        if(rx_counter!=0(
}        y=getchar();}lcd_putchar(' ');lcd_putchar(y؛(
        if(y=='a('
        OCR0=0؛
        else if(y=='b('
        OCR0=100؛
        else if(y=='c('
        OCR0=180؛
        else if(y=='d('
        OCR0=253؛
        else if(y=='p('
}        x='p';break{؛
{
 {
        else if(x=='k('
}        lcd_clear؛()
        lcd_puts("Temp؛("
        y=read_adc(0؛(
        putchar(y؛(
        x='p؛'
        delay_ms(1000؛(
{
```

```
            else if(x=='p('
}           lcd_clear();lcd_puts("Nothing");delay_ms(10؛(
            OCR0=0؛
            PORTB.4=0؛
            PORTB.7=0؛
            PORTB.6=0؛
            PORTB.5=0؛
{
{
{
```

برنامج الماتلاب (كود الواجهة الرسومية) وتبعاً لبروتوكول الاتصال المتفق عليه يجب أن يرسل أي المحارف وبناءً على ذلك نكتب البرنامج التالي :

```
function varargout = prog(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @prog_OpeningFcn, ...
                   'gui_OutputFcn',  @prog_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
   [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
function prog_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;


guidata(hObject, handles);

function varargout = prog_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;


function s1_Callback(hObject, eventdata, handles)
```

```
s=get(handles.s1,'value');
s=round(s);
set(handles.t2,'string',num2str(s));
s2 = serial('COM1', 'BaudRate', 9600);
fopen(s2);
if s==0
fprintf(s2,'%s','a');
elseif s==1
    fprintf(s2,'%s','b');
elseif s==2
    fprintf(s2,'%s','c');
elseif s==3
    fprintf(s2,'%s','d');
end
fclose(s2);
delete(s2);
clear s2;

function s1_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end


function pushbutton3_Callback(hObject, eventdata, handles)
clear all
s2 = serial('COM1', 'BaudRate', 9600);
fopen(s2);
fprintf(s2,'%s','m');
fclose(s2);
delete(s2);
clear s2;

function pushbutton1_Callback(hObject, eventdata, handles)
clear all
s2 = serial('COM1', 'BaudRate', 9600);
fopen(s2);
fprintf(s2,'%s','g');
fclose(s2);
delete(s2);
clear s2;

function pushbutton2_Callback(hObject, eventdata, handles)
```

```
clear all
s2 = serial('COM1', 'BaudRate', 9600);
fopen(s2);
fprintf(s2,'%s','h');
fclose(s2);
delete(s2);
clear s2;

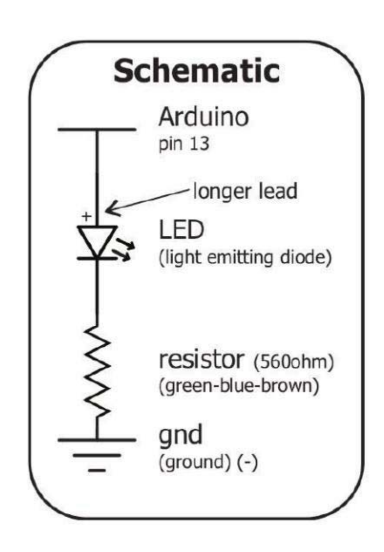

function pushbutton4_Callback(hObject, eventdata, handles)
clear all
s2 = serial('COM1', 'BaudRate', 9600);
fopen(s2);
fprintf(s2,'%s','p');
fclose(s2);
delete(s2);
clear s2;

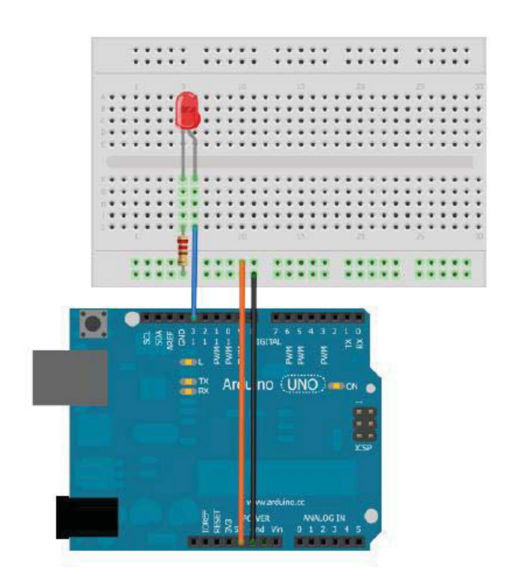

function pushbutton5_Callback(hObject, eventdata, handles)
s2 = serial('COM1', 'BaudRate', 9600);
fopen(s2);
fprintf(s2,'%s','k');
while true
c=fscanf(s2);
if ~isempty(c)
    x=unicode2native(c);
    set(handles.tt,'string',num2str(x));
    break;
end
end
fclose(s2);
delete(s2);
clear s2;
```

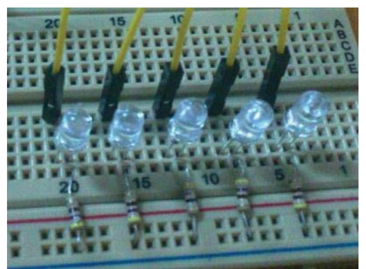
مثال (٣) :

سنقوم فيها بكتابة برنامج للتحكم بإضاءة ليدات من دارة الأردوينو بحيث سنجهز دارة الأردوينو لاستقبال المحارف ASCII عبر النافذة التسلسلية ، و سيتم إرسال هذه المحارف من برنامج الماتلاب.

مخطط الدارة :







برنامج الأردوينو:

```
void setup()
{
  Serial.begin(9600);

//  Set all the pins we need to output pins
  pinMode(2, OUTPUT);
  pinMode(3, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
}

void loop()
{
  if (Serial.available()) {

//    read serial as a character
    char ser = Serial.read();

//    NOTE because the serial is read as "char" and not "int", the read
value must be compared to character numbers
```

```
//    hence the quotes around the numbers in the case statement
    switch (ser) {
      case 'a':
       for(int i=2; i<=6; i+=1)
       triggerPin(i);
        break;
      case 'b':
        for(int i=6; i>=2; i-=1)
        untriggerPin(i);
        break;
      case 'c';
        triggerPin(5);
        untriggerPin(5);
        triggerPin(6);
        untriggerPin(6);
        triggerPin(3);
        untriggerPin(3);
        triggerPin(2);
        untriggerPin(2);
        break;
    }
  }
}

void triggerPin(int pin){
  digitalWrite(pin, HIGH);
  delay(100);
}

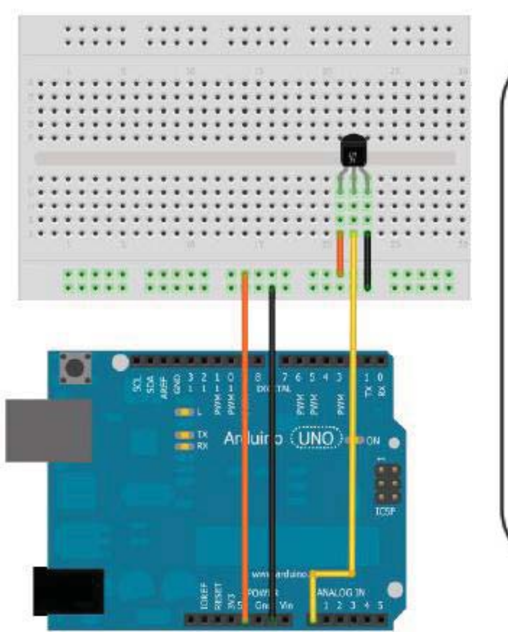
void untriggerPin(int pin){
  digitalWrite(pin, LOW);
  delay(100);
}
```

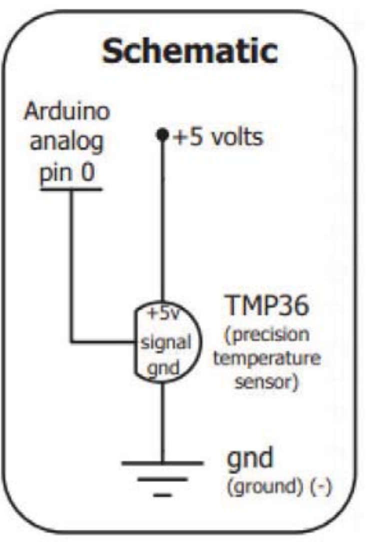
برنامج الماتلاب :

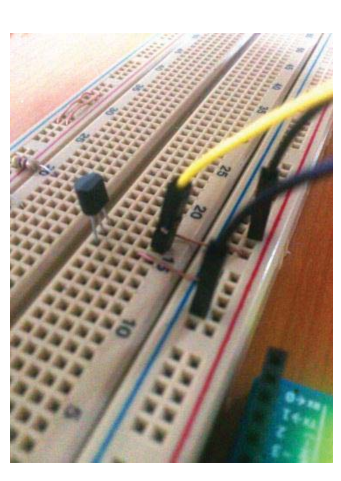
```
s=menu('choose your state :' , '1' , '2' , '3')
s2 = serial('COM1', 'BaudRate', 9600);
fopen(s2);
if s==1
fprintf(s2,'%s','a');
elseif s==2
   fprintf(s2,'%s','b');
elseif s==3
   fprintf(s2,'%s','c');
end
fclose(s2);
delete(s2);
```

مثال (٤) :

سنقوم فيها بكتابة برنامج للقياس درجة الحرارة بالجو بالمحيط باستخدام حساس LM35 و باستخدام دارة الأردوينو بحيث سنجهز دارة الأردوينو لقياس درجة الحرارة و إرسالها للماتلاب لحظياً كل ثانية عبر النافذة التسلسلية ، و سيتم إستقبال قيم درجة الحرارة من الماتلاب كل ثانية و معالجة القيم المدخلة و رسم منحنى تغير درجة الحرارة مع الزمن.







برنامج الأردوينو :

```
const int sensorPin=A0;
int reading;
float voltage;
float temperatureC;
```

```
int value;
void setup()
{
 Serial.begin(9600);
}
void loop()
{
 value=Serial.read();
 reading=analogRead(sensorPin);
 voltage=reading*5.0/1024;
 temperatureC=(voltage-0.5)*100;
 Serial.println(temperatureC);
 delay(1000);
}
```

برنامج الماتلاب :

```
% Find a serial port object.
obj1 = instrfind('Type', 'serial', 'Port', 'COM6', 'Tag', '');

% Create the serial port object if it does not exist
% otherwise use the object that was found.
if isempty(obj1)
    obj1 = serial('COM6');
else
    fclose(obj1);
    obj1 = obj1(1);
end

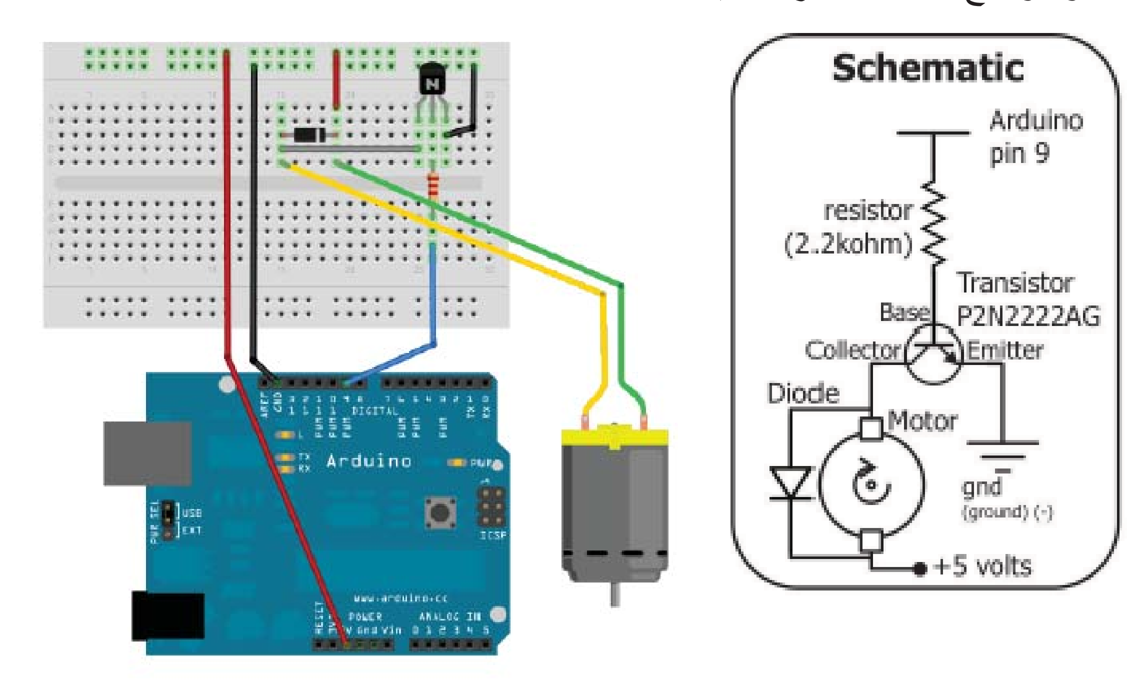
% Connect to instrument object, obj1.
fopen(obj1);

a=menu('Start receive temp. from Serial Port :','Start','No');
if a==1
% Communicating with instrument object, obj1.
for i=1:15
data = fscanf(obj1,'%s');
temp=str2double(data);
disp(temp)
```

```
plot(i,temp,'*g','linewidth',3);
hold on
grid on
pause(1);
end
end
fclose(obj1);
```

مثال (5) :

لنقوم فيما يلي بتجهيز بروتوكول اتصال بين الأردوينو و الماتلاب للتحكم بمحرك تيار مستمر كما هو موضح بمخطط الدارة المبينة :



برنامج الأردوينو :

```
int motorPin=9;
int onTime=5000;
int offTime=2000;

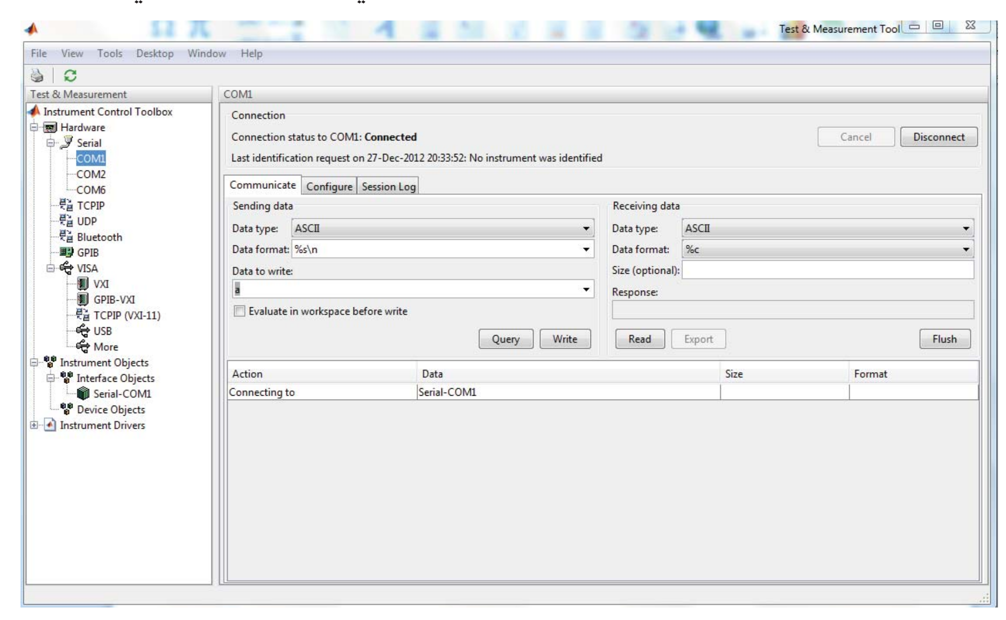
void setup()
{
 Serial.begin(9600);
 pinMode(2, OUTPUT);
 pinMode(motorPin,OUTPUT);
}
```

```
void loop()
{
  if (Serial.available()) {
  //read serial as a character
  char ser = Serial.read();
   switch (ser) {
 case 'a':
 analogWrite(motorPin,100);
 delay(onTime);
 digitalWrite(motorPin,LOW);
 delay(offTime);

 case 'b':
 analogWrite(motorPin,200);
 delay(onTime);
 digitalWrite(motorPin,LOW);
 delay(offTime);
    }
  }
}
```

سنقوم بالاستفادة من الأداة tmtool الربط مع المنفذ التسلسلي و ذلك كما يوضح الشكل التالي .

ملاحظة : يمكن الوصول للأداة المذكورة بكتابة tmtool في سطر الأوامر في الماتلاب.

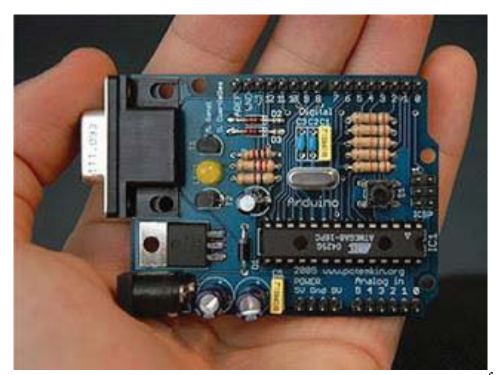


التحكم بالأردوينو

Controlling Arduino Board

١٠,١٠ مـا هـو أردويـنـو ؟

- Arduino عبارة عن دارة الكترونية تقوم بتيسير برمجة متحكم صغري Microcontrollerمن نوع Atmel عن طريق الحاسوب .
- لوحة مطبوعة تم انتاجها في إيطاليا سنة ٢٠٠٥ كحل لمشاريع الطلاب، وتطورت شيئاً فشيئاً وحالياً يتم تصنيعها من قبل شركة تسمى Smart Projects.
- يوجد عليها متحكم من شركة Atmel وبه عدد من المداخل والمخارج الرقمية والتماثلية، وهناك مدخل USB ليتم برمجة المتحكم من خلاله عند وصل اللوحة بالحاسب. وكل المداخل والمخارج متصلة بأرجل Pins سهلة الوصول كي تسهل على المستخدم استغلالها في تصاميمه .



وهي تسمح بنوع متطور من أنواع الإحساس والتفاعل مع العالم الخارجي أكثر مما يسمح به حاسوبك المكتبي. حيث بإمكانك استخدامها لتلقي إشارات مختلفة من مجموعة متنوعة من الحساسات ، وكذلك إرسال اوامر لمجموعة كبيرة من العناصر الخارجية مثل المحركات والقواطع والليدات.

هذه المتحكمة لها عدة مميزات تميزها عن غيرها من المتحكمات الصناعية :

- **مفتوحة المصدر**

حيث قام مجموعة من المهندسين من إيطاليا بتصميم هذه الدارة وتوزيعها بشكل مفتوح المصدر أي يمكن لأي شخص التعديل عليها (كما يحدث للبرامج مفتوحة المصدر) واستعمالها في كافة المشاريع. هذه الدارة تعتمد على معالج ATMEGA8 أو ATMEGA168 كمتحكمة والمسؤولة عن تلقي ومعالجة الاوامر وارسال الاشارات (حاسوب صغير).

- **سهلة الاستخدام**

ويمكن تعلمها بسهولة وهي مدعومة من قبل الكثير من هواة ومحترفي الالكترونيات والهاكرز في الكثير من المشاريع.

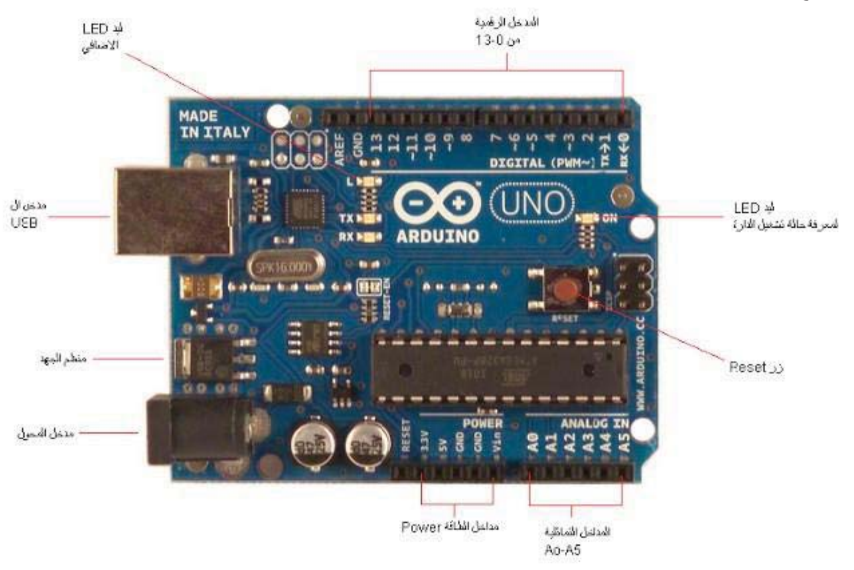
- **لغة خاصة**

لها لغة برمجة خاصة وسهلة ومفتوحة المصدر وتعمل على أغلب أنظمة التشغيل، أيضا مما يميز لغة البرمجة هذه أنك ستجد العديد من الاشخاص الذين أوجدوا مكتبات لربط هذه اللغة مع لغات مختلفة مثل Matlab و VS.NET وغيرها من اللغات لذا فهناك بحر من الخيارات. .

رخيصة الثمن
تبدأ التكلفة من ٣٠ دولار (مع تكاليف الشحن)،وبإمكاننا تصنيعها بأنفسنا إذا كنا نملك المواد والادوات المناسبة لذلك.

أردوينـو UNO

- الـ Arduino Uno عبارة عن دارة ميكروكونترولر تعتمد على معالج الاتميل ATmega328
- تحوي هذه الدارة على ١٤ مدخل/مخرج من النوع الرقمي (Digital) من هذه الـ ١٤ يوجد ٦ يمكن استخدامها كمخارج PWM أو ما يعرف بالتعديل الرقمي المعتمد على عرض النبضة (Pulse-Width modulation).
- أيضا تحوي الدارة ٦ مداخل تماثلية Analog ، ومهتز كريستال بتردد ١٦MHz، مدخل USB من أجل التواصل مع الحاسب، مدخل طاقة، و ICSP header والذي يعني القدرة على برمجة المتحكمة وهي لا تزال موصلة بالعتاد وهذا يوفر الكثير من الوقت والجهد مما يغني عن فك الدارة وتوصيلها بمبرمجة خاصة ومن ثم تركيبها على الدارة مرة أخرى.
- هذه المتحكمة تحوي كل ما تحتاج لكي تعمل سواء عن طريق منفذ ال USB أو عن طريق المحول مباشرة.



تتم برمجة المتحكم عن طريق بيئة التطوير الخاصة بالأردوينو IDE و التي يمكن تحميلها من الموقع الرسمي مباشرةً.
وقد تم ترجمة واجهة هذه البيئة تشاركياً إلى اللغة العربية.

تم مؤخراً تطوير مجموعة من الحزم الداعمة لدارة الأردوينو و التي تسمح بالتحكم بها و برمجتها بشكل مباشر من الماتلاب ، و تؤمن هذه الحزم مجموعة واسعة من الميزات التي تسمح برفع أداء النظام و الكثير من الميزات الأخرى.

١١,١٠ الحزم الداعمة Support Packages

- حزمة الإدخال و الإخراج الرقمي Arduino IO Package

تستخدم لأداء الإدخال و الإخراج الرقمي و التماثلي و التحكم بالمحركات من خلال سطر الأوامر في الماتلاب.

- حزمة التحكم بالمحركات Arduino Motor Package

تستخدم لأداء عمليات التحكم بمختلف أنواع المحركات (محركات التيار المستمر – المحركات الخطوية – محركات السيرفو) مع تأمين حزمة واسعة من التوابع اللازمة لعمليات الإدخال و الإخراج الرقمي و التماثلي.

١٢,١٠ تجهيز الأردوينو و إعداد الماتلاب

١) قم بتحميل حزمة الدعم من الرابط : هنا

٢) تجهيز الأردوينو للإدخال و الإخراج الرقمي و التماثلي :

يتم ذلك بتحميل ملف adiosrv.pde على دارة الأردوينو من خلال برنامج Arduino (بيئة التطوير الخاصة بالأردوينو IDE).

٣) تجهيز الماتلاب يتم بإضافة التوابع الخاصة بالأردوينو للمكتبة .

١٣,١٠ التحكم بالأردوينو من الماتلاب مقابل التحكم بالطريقة التقليدية IDE Environment

MATLAB vs. IDE Environment

- الماتلاب أكثر فاعلية ، و يمكن تنفيذ أوامر الإدخال و الإخراج لحظياً دون الحاجة للبرمجة، الترجمة، التحميل و التنفيذ كل مرة.
- الكتابة في الماتلاب أكثر سهولة للفهم من الكتابة بالـ C (التعامل مع بيانات مختلفة، الأشعة، سطور برمجية أقل) و ذلك يعني :
  - الماتلاب أنسب للمشاريع المعقدة.
  - إمكانية التنفيذ بسرعة أكبر في الماتلاب.
- الماتلاب مناسب للمشاريع الواسعة التي قد تحتاج لتحليل النتائج ، معالجة الإشارة، الحسابات الرياضية المعقدة، المحاكاة، الإحصاء، و تصميم نظم التحكم ..... إلخ

١٤,١٠ التحكم بالأردوينو من البيئة البرمجية في الماتلاب

- إنشاء الاتصال arduino('port')

استخدم الأمر a=arduino('port') مع تحديد الاسم الصحيح للمنفذ كمتحول نصي، ليبدأ الاتصال بين الماتلاب و البورد و تنشأ الكائن الخاص بالأردوينو في workspace:

```
>> a=arduino('com6);
```

- تعيين نمط الأرجل a.pinMode

استخدم الأمر a.pinMode(pin,str) لتحصل أو تضبط نمط الأرجل pin لتكون دخل أو خرج.

```
>> a.pinMode(11,'output');
>> a.pinMode(10,'input');
>> val=a.pinMode(10);
>> a.pinMode(5);
>> a.pinMode;
```

- القراءة و الكتابة الرقمية

استخدم الأمر a.digitalRead(pin) للقراءة الرقمية من الـ pin :

```
>> val=a.digitalRead(4)
```

وسيعيد التابع قيمة رقمية منطقية (0 or 1) .

استخدم الأمر a.digitalWrite(pin,val) للكتابة الرقمية على الـ pin :

```
>> a.digitalWrite(13,1);   # High
>> a.digitalWrite(13,0);   # Low
```

- القراءة و الكتابة التماثلية

استخدم الأمر a.analogRead(pin) للقراءة التماثلية من الـ pin :

```
>> val=a.analogRead(0)
```

وسيعيد التابع قيمة ضمن المجال (0 – 1023) .

استخدم الأمر a.analogWrite(pin,val) للكتابة الرقمية على الـ pin و مجال القيمة val محدود ضمن مجال (0 – 255).

```
>> a.analogWrite(11,90);   # set pin #11 to 90
>> a.analogWrite(3,10);
```

- قطع الاتصال

استخدم الأمر delete(a) للقطع الاتصال مع الـ pin :

```
>> delete(a);
```

بعد قطع الاتصال المنفذ التسلسلي الخاص ببورد الأردوينو يصبح متاح للاتصال عن طريق بيئة اخرى .

مثال (١) :

التحكم بإضاءة ليدات بحيث يعمل كل ليد من الليدات على المخارج رقم (٨-٩-١٠-١١-١٢) بشكل متسلسل و بتأخير زمني ثانية واحدة.



```
a=arduino('com6');
for i=8:13
  a.pinMode(i,'output');
  a.digitalWrite(i,0);
  pause(1);
end
delete(a);
```

مثال (٢) :

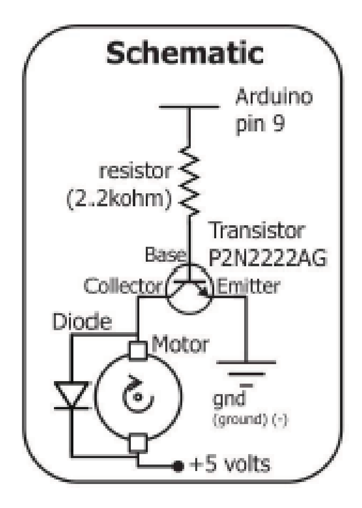
برنامج لقراءة قيم درجة الحرارة و رسم تغيراتها مع الزمن ، يستخدم في هذا المثال الحساس LM35.




```
a=arduino('com6');
for t=1:15
reading=a.analogRead(3);
voltage=reading*5.0/1024;
temperatureC=(voltage-0.5)*100;
plot(t,temperatureC,'*g');
hold on
disp('Temperature is :');
disp(temperatureC);
pause(1);
end
delete(a);
```

مثال (٣) :

برنامج للتحكم بمحرك تيار مستمر .





```
a=arduino('com6');
a.pinMode(9,'output');
for i=1:5
    a.analogWrite(9,100);
    pause(5);
    a.digitalWrite(9,0)
    pause(2)
end
delete(a);
```



تحليل الدارات الكهربائية

- تحليل دارات التيار المستمر .
- تحليل دارات التيار المتناوب .
- دراسة الدارات الكهربائية في الحالة العابرة .
- الأنظمة ثلاثية الطور .

تحليل الدارات الكهربائية

١,١١ : تحليل دارات التيار المستمر

١,١,١١ مقدمة

في هذا الفصل سنعتمد على الطرائق المختلفة المعتمدة في حساب التيارات والتوترات في دارات التيار المستمر ، ومن ثم سنعتمد على برنامج الماتلاب من أجل حل التوابع المعقدة التي يصعب الوصول لنتائجها بدون الحاسب.

٢,١,١١ تحليل نقاط الدارة (قانون كيرشوف الأول)

$$Y_{11}V_1 + Y_{12}V_2 + ...+ Y_{1m}V_m = \sum I_1$$

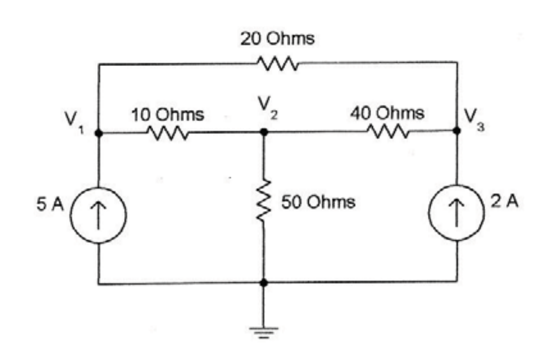
$$Y_{21}V_1 + Y_{22}V_2 + ...+ Y_{2m}V_m = \sum I_1$$

$$Y_{m1}V_1 + Y_{m2}V_2 + ...+ Y_{mm}V_m = \sum I_m$$

$$[Y][V] = [I] \quad \rightarrow \quad [V] = [Y]^{-1}[I]$$

مثال :

المطلوب إيجاد كمونات العقد للدارة المبينة بالشكل التالي



بتطبيق قانون كيرشوف على العقد نجد

العقدة ١ : $0.15V_1 - 0.1V_2 - 0.05V_3 = 5 \quad \leftarrow \quad \frac{V_1-V_2}{10} + \frac{V_1-V_3}{20} - 5 = 0$

العقدة ٢ : $-0.1V_1 + 0.145V_2 - 0.025V_3 = 0 \quad \leftarrow \quad \frac{V_2-V_1}{10} + \frac{V_2}{50} + \frac{V_2-V_3}{40} = 0$

العقدة ٣ : $\frac{V_3 - V_1}{20} + \frac{V_3 - V_2}{40} - 2 = 0 \quad \rightarrow \quad -0.05V_1 - 0.025V_2 + 0.075V_3 = 3$

$$\begin{bmatrix} 0.15 & -0.1 & -0.05 \\ -0.1 & 0.145 & -0.025 \\ -0.05 & -0.025 & 0.075 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \\ 2 \end{bmatrix}$$

ويكون الكود البرمجي الذي نكتبه في الماتلاب كملف M-file كالتالي ....

```
clc
clear
Y=[0.15 -0.1 -0.05; -0.1 0.145 -0.025; -0.05 -0.025 0.075];
I=[5; 0; 2];
fprintf('Nodal Voltages V1, V2 and V3 are: \n')
v=inv(Y)*I
```

النتائج التي حصلنا عليها بعد التطبيق :

```
Nodal Voltages V1, V2 and V3 are:
v =
    404.2857
    350.0000
    412.8571
```

٣,١,١١ تحليل حلقات الدارة (قانون كيرشوف الثاني)

$$Z_{11}I_1 + Z_{12}I_2 + Z_{13}I_3 + \ldots + Z_{1n}I_n = \sum V_1$$

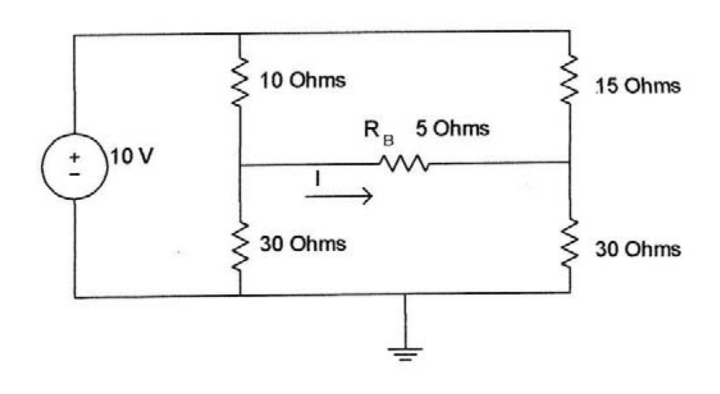
$$Z_{21}I_1 + Z_{22}I_2 + Z_{23}I_3 + \ldots + Z_{2n}I_n = \sum V_2$$

$$Z_{n1}I_1 + Z_{n2}I_2 + Z_{n3}I_3 + \ldots + Z_{nn}I_n = \sum V_n$$

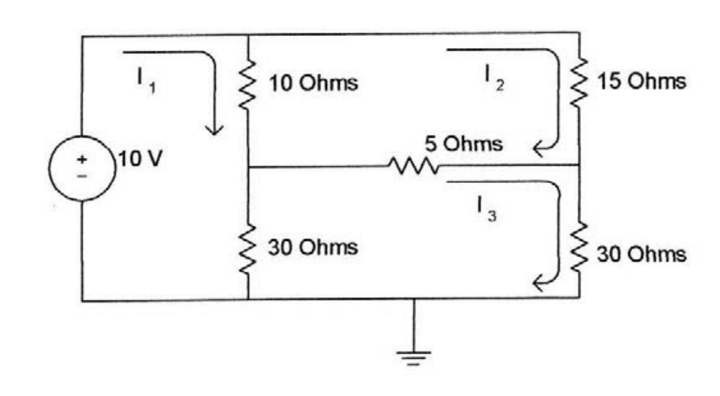
$$[Z][I] = [V] \quad \rightarrow \quad [I] = [Z]^{-1}[V]$$

مثال :

المطلوب حساب قيمة التيار المار عبر المقاومة R_B ومن ثم إيجاد الاستطاعة التي يؤمنها منبع الجهد (10v) ، وذلك للدارة الموضحة بالشكل الموضح



بتحليل الحلقات كما يبين الشكل التالي :



الحلقة ١ : $40I_1$ - $10I_2$ - $30I_3$ = 10 → $10(I_1 - I_2) + 30(I_1 - I_3) - 10 = 0$

الحلقة 2 : $-10I_1 + 30I_2 - 5I_3 = 0$ → $10(I_2 - I_1) + 15I_2 + 5(I_2 - I_3) = 0$

الحلقة 3 : $-30I_1 - 5I_2 + 65I_3 = 0$ → $30(I_3 - I_1) + 5(I_3 - I_2) + 30I_3 = 0$

$$\begin{bmatrix} 40 & -10 & -30 \\ -10 & 30 & -5 \\ -30 & -5 & 65 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 10 \\ 0 \\ 0 \end{bmatrix}$$

ويكون الكود البرمجي الذي نكتبه في الماتلاب كملف M-file كالتالي ....

```
clear
clc
Z=[40 -10 -30; -10 30 -5; -30 -5 65];
V=[10; 0; 0];
I=inv(Z)*V;
IRB=I(3)-I(2);
```

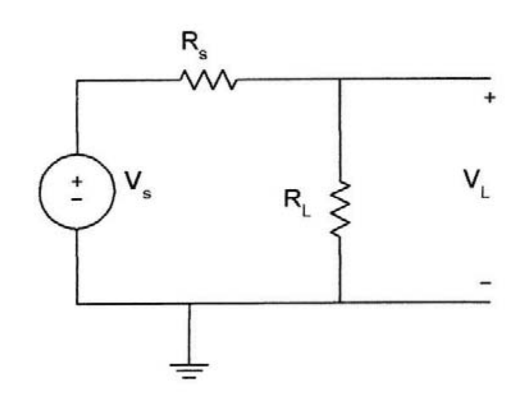
```
fprintf('The current through R is %8.10f Amps \n', IRB)
PS=I(1)*10;
fprintf('The Power supplied by 10V source is %8.10f watts \n', PS)
```

النتائج التي حصلنا عليها بعد التطبيق :

```
The current through R is 0.0370370370 Amps
The Power supplied by 10V source is 4.7530864198 watts
```

٤,١,١١ الاستطاعة العظمى المنقولة

لنفرض لدينا منبع الجهد المبين بالشكل الموضح حيث R_s مقاومة منبع الجهد و R_L الحمل



باستخدام مجزئ الجهد نجد أن : $V_L = \frac{V_S.R_L}{R_S+R_L}$ & $P_L = \frac{V_L^2}{R_L} = \frac{V_S^2.R_L}{(R_S+R_L)^2}$

وللحصول على القيمة مقاومة الحمل التي تعطينا القيمة الأعظمية للطاقة نشتق معادلة الاستطاعة الأخيرة بالنسبة للمقاومة R_L ونجعل الناتج مساوي للصفر فنجد :

$$\frac{dP_L}{dR_L} = \frac{(R_S+R_L)^2 V_S - V_S^2 R_L(2)(R_S+R_L)}{(R_S+R_L)^4} \quad \rightarrow \quad \frac{dP_L}{dR_L} = 0$$

وبتبسيط العلاقة الأخيرة بعد جعلها مساوية للصفر نجد : $R_L = R_S$

ويمكن استخدام الماتلاب لمراقبة تغيير الجهد وتبديد الطاقة في الحمولة وذلك تبعاً لقيمة مقاومة الحمل.

ملاحظة

'Find' function

يستخدم هذا التابع لتحديد قيم العناصر من المصفوفة التي قيمها لا تساوي الصفر

مثال :

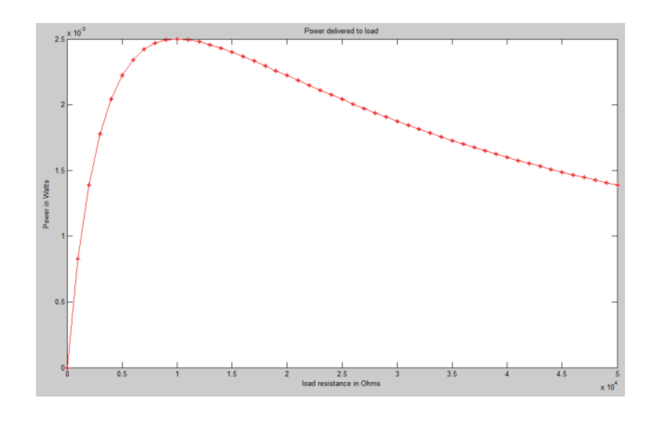
في الدارة الأخيرة لنفرض أن قيمة مقاومة الحمل تتغير من ٠ إلى ٥٠كيلواوم ، والمطلوب رسم تغييرات الاستطاعة في الحمل ، حدد الاستطاعة العظمى على الحمل عندما R_L=10KΩ.

الكود البرمجي الذي نكتبه في الماتلاب كملف M-file كالتالي :

```
clc
clear
vs=10;   rs=10e3;      rl=0:1e3:50e3;
k=length(rl);
for i=1:k
   pl(i) = ((vs/(rs+rl(i)))^2)*rl(i);
end
dp=diff(pl)./diff(rl);
rld=rl(2:length(rl));
prod=dp(1:length(dp)-1).*dp(2:length(dp));
crit_pt=rld(find(prod<0));
max_power=max(pl);
fprintf('Maximum Power occurs at %8.3f ohms  \n', crit_pt)
fprintf('Maximum Power dissipation is %8.5f Watts  \n', max_power)
plot(rl,pl,'r-*')
title('Power delivered to load')
xlabel('load resistance in Ohms')
ylabel('Power in Watts')
```

النتائج التي حصلنا عليها بعد التطبيق :

```
Maximum Power occurs at 10000.000 ohms
Maximum Power dissipation is  0.00250 Watts
```

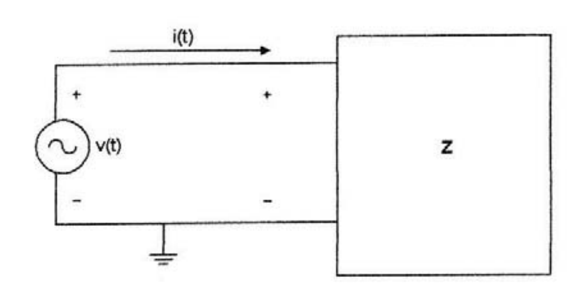


٢,١١ تحليل دارات التيار المتناوب

١,٢,١١ مقدمة

في هذه الجلسة سندرس دارات التيار المتناوب وسنستعين بالتكامل العددي للحصول على القيم الوسطية للاستطاعة. سنتطرق بعد ذلك لتحليل الدارات ثلاثية الطور وذلك بتحويلها لمجال التردد ومن ذلك الاستعانة بقوانين كيرشوف لحل الدارة. ولما سبق سنعتمد على المصفوفات من أجل حسابات التوترات والتيارات. وسنستعين بتوابع كثيرة في الماتلاب سيكون لها دور كبير في تسهيل الحسابات.

٢,٢,١١ دراسة الحالة المستقرة لدارات التيار المتناوب



$v(t) = V_m.cos(wt + \theta_V)$ $\quad$ $i(t) = I_m.\cos(wt + \theta_I)$

$V_{rms}=\sqrt{\frac{1}{T}\int_0^T v^2(t)dt}$ $\quad$ $I_{rms}=\sqrt{\frac{1}{T}\int_0^T i^2(t)dt}$

$V_{rms}=\frac{V_m}{\sqrt{2}}$ $\quad$ $I_{rms}=\frac{I_m}{\sqrt{2}}$

The average power dissipated :

$$P = \frac{1}{T}\int_0^T v(t)i(t)dt = V_{rms}I_{rms}\cos(\theta_V - \theta_I)$$

The power factor :

$$pf = \frac{P}{V_{rms}.I_{rms}} = cos(\theta_V - \theta_I)$$

The complex power, *S*, is :

$$S = P + jQ = V_{rms}.I_{rms}.[\cos(\theta_V - \theta_I) + j\sin(\theta_V - \theta_I)]$$

ملاحظة

'quad & quad8' functions

يوفر الماتلاب هاذين التابعين من أجل إجراء التكامل لأي تابع والصيغة العامة للتابعين هي :

$$q = \int_a^b funct(x)dx$$

quad('funct',a,b,tol,trace)

quad8('funct',a,b,tol,trace)

حيث :

funct	اسم التابع كما تم تعريفه في الماتلاب
a	القيمة الحدية الدنيا للتكامل
b	القيمة الحدية العليا للتكامل
tol	حدود التأرجح المسموح بها من أجل الدقة ولها قيمة افتراضية
trace	لتفعيل إمكانية رسم التكامل ويمكن تفعيلها بوضع أي قيمة غير مساوية للصفر حيث أن القيمة صفر هي الافتراضية

مثال :

$v(t) = 10.\cos(120\pi t + 30)$ $\qquad$ $i(t) = 6.\cos(120\pi t + 60)$

والمطلوب تحديد القيمة الوسطية للطاقة والقيمة اللحظية للتوتر وكذلك معامل الاستطاعة وذلك بالطريقتين التحليلية والرياضية.

من أجل ذلك نكتب الكود البرمجي التالي في ملف M-file كالتالي :

```
clc
T=2*pi/(120*pi);
a=0;
b=T;
x=0:0.02:1;
t=x.*b;
v_int=quad('voltage1',a,b);
```

```
v_rms=sqrt(v_int/b);
i_int=quad('current1',a,b);
i_rms=sqrt(i_int/b);
p_int=quad('inst_pr',a,b);
p_ave=p_int/b;
pf=p_ave/(i_rms*v_rms);
p_ave_an=(60/2)*cos(30*pi/180);
v_rms_an=10/sqrt(2);
pf_an=cos(30*pi/180);
fprintf('Average power, analytical: %f \n average power, numerical: %f
\n', p_ave_an, p_ave)
fprintf('rms power, analytical: %f \n rms power, numerical: %f \n',
p_ave_an, p_ave)
fprintf('Power vactor, analytical: %f \n power factor, numerical: %f \n',
pf_an, pf)
```

نلاحظ في البرنامج أننا استخدمنا عدة توابع ولذلك لابد من كتابتها في ملفات خاصة بها وهي كالتالي (يكتب كل تابع في ملف لوحده)

```
function vsq=voltage1(t)
vsq=(10*cos(120*pi*t + 60*pi/180)).^2;
end

function isq=current1(t)
isq=(6*cos(120*pi*t+30*pi/180)).^2;
end

function pt=inst_pr(t)
it=6*cos(120*pi*t+30*pi/180);
vt=10*cos(120*pi*t+60*pi/180);
pt=it.*vt;
end
```

النتائج التي حصلنا عليها بعد التطبيق :

```
Average power, analytical: 25.980762
average power, numerical: 25.980762
rms power, analytical: 25.980762
rms power, numerical: 25.980762
Power vactor, analytical: 0.866025
```

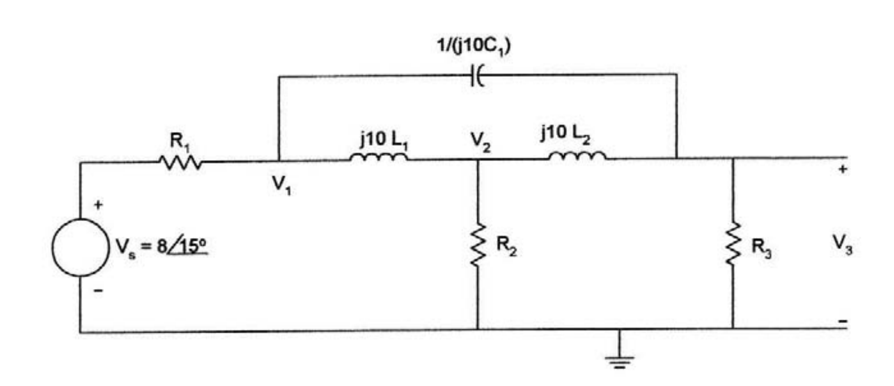
power factor, numerical: 0.866025

وبالتالي نجد أنه باستخدام الطريقة التحليلية أو الرقمية فإن النتائج ستكون متشابهة تماماً.

٣,٢,١١ دارات التيار المتناوب الأحادية والثلاثية الطور

مثال (1)

لتكن لدينا الدارة الموضحة بالشكل ...



R_1=20 Ω R_2=100 Ω R_3=50 Ω

L_1= 4 H L_2= 8 H C_1 = 250µf

w = 10 rad/s

والمطلوب حساب طويلة وزاوية توتر الخرج .....

العقدة ١ :

$$\frac{(V_1 - V_S)}{R_1} + \frac{(V_1 - V_2)}{j10L_1} + \frac{(V_1 - V_3)}{1/(j10C_1)} = 0$$

العقدة ٢ :

$$\frac{(V_2 - V_1)}{j10L_1} + \frac{V_2}{R_2} + \frac{(V_2 - V_3)}{j10L_2} = 0$$

العقدة ٣ :

$$\frac{V_3}{R_3} + \frac{(V_3 - V_2)}{j10L_2} + \frac{(V_3 - V_1)}{1/j10C_1} = 0$$

وبعد ترتيب عناصر المعادلات الثلاثة الأخيرة وإعادة تنسيقها نحصل على المصفوفة التالية ...

$$\begin{bmatrix} 0.05 - j0.0225 & j0.025 & -0.0025 \\ j0.025 & 0.01 - j0.0375 & j0.0125 \\ -j0.0025 & j0.0125 & 0.02 - j0.01 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 0.4\angle 15 \\ 0 \\ 0 \end{bmatrix}$$

برنامج حل المصفوفة ....

```
clc
clear
Y=[0.05-0.0225*j 0.025*j -0.0025*j; 0.025*j 0.01-0.0375*j 0.0125*j; -
0.0025*j 0.0125*j 0.02-0.01*j];
c1=0.4*exp(pi*15*j/180);
I=[c1; 0; 0];
V=inv(Y)*I;
v3_abs=abs(V(3));
v3_ang=angle(V(3))*180/pi;
fprintf('Voltage V3, magnitude : %f \nvoltage V3, angle in degree : %f
\n', v3_abs,v3_ang)
```

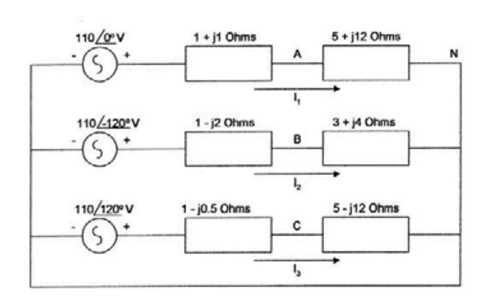
النتائج التي حصلنا عليها بعد التطبيق :

Voltage V3, magnitude : 1.850409
voltage V3, angle in degree : -72.453299

$$v_3(t) = 1.85\cos(10t - 72.45) \text{ v}$$

مثال (2)

ليكن لدينا النظام غير المتوازن المبين بالشكل التالي ...



والمطلوب حساب التوترات الطورية V_{AN} , V_{BN} , V_{CN}.

باستخدام قانون كيرشوف نجد :

$$110\angle 0 = (1 + j)I_1 + (5 + 12j)I_1 \quad \rightarrow \quad 110\angle 0 = (6+13j)I_1$$

110∠-120 = (1 – 2j)I_2 + (3 + 4j)I_2 → 110∠-120 = (4 + 2j)I_2

110∠120 = (1 – 0.5j)I_3 + (5 – 12j)I_3 → 110∠120 = (6 – 12.5j)I_3

$$\begin{bmatrix} 6+j13 & 0 & 0 \\ 0 & 4+2j & 0 \\ 0 & 0 & 6-j12.5 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 110\angle 0 \\ 110\angle -120 \\ 110\angle 120 \end{bmatrix}$$

$[I]$ = inv(Z)*V

V_{AN} = (5 + 12j)I_1

V_{BN} = (3 + 4j) I_2

V_{CN} = (5 - 12j)I_3

برنامج الماتلاب ...

```
clc
clear
Z=[6-13*j 0 0; 0 4+2*j 0; 0 0 6-12.5*j];
c2=110*exp(j*pi*(-120/180));
c3=110*exp(j*pi*(120/180));
V=[110; c2; c3];
I = inv(Z)*V;
Van=(5+12*j)*I(1);
Vbn=(3+4*j)*I(2);
Vcn=(5-12*j)*I(3);
Van_abs=abs(Van);
Van_ang=angle(Van)*180/pi;
Vbn_abs=abs(Vbn);
Vbn_ang=angle(Vbn)*180/pi;
Vcn_abs=abs(Vcn);
Vcn_ang=angle(Vcn)*180/pi;
fprintf('Phasor voltage Van, magnitude : %f \nphasor Voltage Van,
angle in degree : %f \n', Van_abs, Van_ang)
fprintf('Phasor voltage Vbn, magnitude : %f \nphasor Voltage Vbn,
angle in degree : %f \n', Vbn_abs, Vbn_ang)
```

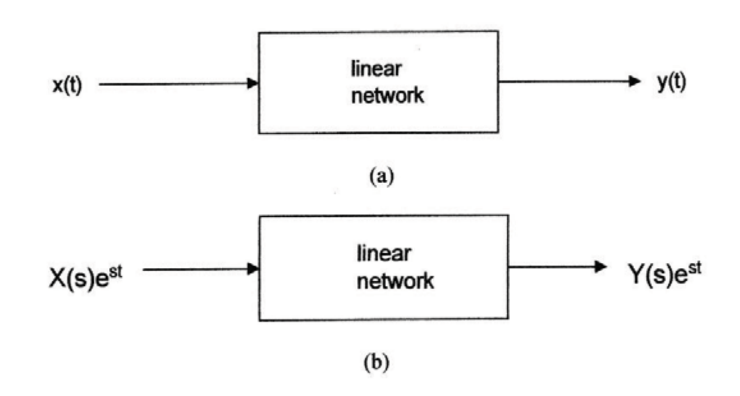
```
fprintf('Phasor voltage Vcn, magnitude : %f \nphasor Voltage Vcn, angle in degree : %f \n', Vcn_abs, Vcn_ang)
```

النتائج التي حصلنا عليها بعد التطبيق :

```
Phasor voltage Van, magnitude : 99.875532
phasor Voltage Van, angle in degree : 132.604994
Phasor voltage Vbn, magnitude : 122.983739
phasor Voltage Vbn, angle in degree : -93.434949
Phasor voltage Vcn, magnitude : 103.134238
phasor Voltage Vcn, angle in degree : 116.978859
```

٤,٢,١١ مميزات الشبكة

الشكل يبين شبكة خطية لها دخل x(t) وخرج y(t) .



يمكن تمثيل هذه الدخل والخرج في هذه الشبكة بالمعادلة التفاضلية

$$a_n \frac{d^n y(t)}{dt^n} + a_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \ldots.. + a_1 \frac{dy(t)}{dt} + a_0 y(t) = b_m \frac{d^m x(t)}{dt^m} + b \frac{d^{m-1} x(t)}{dt^{m-1}} + \ldots.. + b_1 \frac{dx(t)}{dt} + b_0 x(t)$$

حيث :

$a_n, a_{n-1}, \ldots\ldots, a_0, b_m, b_{m-1}, \ldots\ldots, b_0$ ثوابت حقيقية

بالانتقال إلى مستوي لابلاس تصبح المعادلة التفاضلية

$$(a_n s^n + a_{n-1} s^{n-1} + \ldots\ldots + a_1 s^1 + a_0) Y(s) e^{st} = (b_m s^m + b_{m-1} s^{m-1} + \ldots\ldots + b_1 s^1 + b_0) X(s) e^{st}$$

وبالتالي فإن التابع الذي يعبر عن الشبكة الخطية يكون بالشكل

$$H(s) = \frac{Y(s)}{X(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \ldots\ldots + b_1 s^1 + b_0}{a_n s^n + a_{n-1} s^{n-1} + \ldots\ldots + a_1 s^1 + a_0}$$

وبإعادة ترتيب تابع الشبكة نحصل على الشكل التالي

$$H(s) = \frac{k(s-z_1)(s-z_2)\ldots\ldots(s-z_m)}{(s-p_1)(s-p_2)\ldots\ldots(s-p_n)} = \frac{r_1}{s-p_1} + \frac{r_2}{s-p_2} + \ldots\ldots + \frac{r_n}{s-p_n} + k(s)$$

حيث :

$p_1, p_2 \ldots\ldots, p_n$: أقطاب تابع الشبكة.

$z_1, z_2 \ldots\ldots, z_m$: رواسب الشبكة.

ملاحظة :

نستعمل تفريق الكسور في تبسيط التوابع الكسرية للحصول على الأقطاب والرواسب ونستخدم لذلك التابع *residue* والذي له الصيغة العامة :

[*r*,*p*,*k*] = *residue*(*num*,*den*)

مثال (1)

$$H(s) = \frac{4s^4+3s^3+6s^2+10s+20}{s^4+2s^3+5s^2+2s+8}$$

من أجل تفريق الكسر المبين نكتب مايلي :

```
num=[4 3 6 10 20];
den=[1 2 5 2 8];
[r,p,k]=residue(num,den)
```

النتائج التي حصلنا عليها عند تنفيذ البرنامج :

```
r =

  -1.6970 + 3.0171i
  -1.6970 - 3.0171i
  -0.8030 - 0.9906i
  -0.8030 + 0.9906i


p =

  -1.2629 + 1.7284i
  -1.2629 - 1.7284i
   0.2629 + 1.2949i
   0.2629 - 1.2949i


k =

     4
```

ويمكن العودة بشكل معاكس للتابع الأصلي بكتابة :

[*num*,*den*] = residue[*r*,*p*,*k*]

وستكون النتيجة في نسقين منفصلين ، الأول يمثل معاملات كثير الحدود (البسط) والثاني معاملات كثير الحدود (المقام).

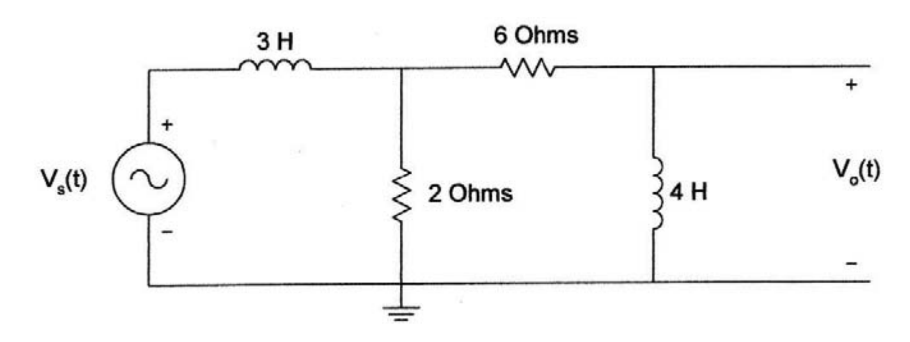
مثال (٢)

للدارة المبينة بالشكل :

١- أوجد تابع الشبكة $H(s) = \frac{V_O(s)}{V_S(s)}$.

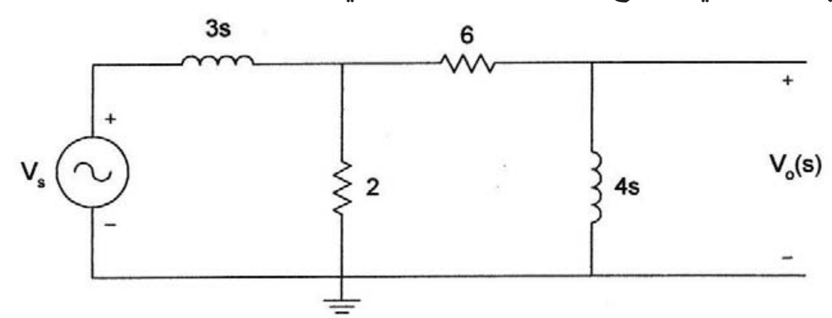
٢- أوجد أقطاب ورواسب تابع الشبكة $H(s)$.

٣- إذا كان $v_s(t) = 10e^{-3t}\cos(2t + 40)$ فأوجد $v_o(t)$.



الحل :

بالانتقال للمستوي اللابلاسي تصبح عناصر الدارة كالتالي



$$\frac{V_o(s)}{V_s(s)} = \frac{V_o(s)V_X(s)}{V_X(s)V_s(s)} = \frac{4s}{(6+4s)}\frac{[2||(6+4s)]}{[2||(6+4s))+3s]} = \frac{4s^2+6s}{6s^3+25s^2+30s+9}$$

تذكر أن :

$$V = V_m\angle\theta \quad \rightarrow \quad V_s = 10\angle 40$$

$$s = \sigma + jw \quad \rightarrow \quad s = -3 + 2j$$

$$V_0(s) = (10\angle 40)H(s)|_{s=-3+2j}$$

والآن سنستخدم البيئة البرمجية (ماتلاب) من أجل إيجاد الأقطاب والرواسب للتابع

```
clear
clc
num=[4 6 0];
den=[6 25 30 9];
disp('The zeros are : ')
z=roots(num)
disp('The poles are : ')
p=roots(den)
s1=-3+2*j;
n1=polyval(num,s1);
d1=polyval(den,s1);
vo=10*exp(j*pi*(40/180))*n1/d1;
```

```
vo_abs=abs(vo);
vo_ang=angle(vo)*180/pi;
fprintf('phasor voltage vo, magnitude : %f \nphasor voltage vo, angle in degrees : %f \n' , vo_abs, vo_ang)
```

النتائج التي حصلنا عليها بعد التطبيق :

```
The zeros are :
z=
٠
-١,٥٠٠٠

The poles are :
p=
-٢,٢١٥٣
-١,٥٠٠٠
-٠,٤٥١٤
phasor voltage vo, magnitude : 3.453492
phasor voltage vo, angle in degrees : -66.990823
```

وبالنتيجة يكون الخرج :

$$v_o(t) = 3.45e^{-3t}\cos(2t - 66.99)$$

٥,٢,١١ الاستجابة الترددية

تعطى الصيغة العامة لتابع النقل لإشارة تمثيلية بالمعادلة :

$$H(s) = \frac{Y(s)}{X(s)} = \frac{b_m s^m + b_{m-1}s^{m-1} + \ldots\ldots + b_1 s^1 + b_0}{a_n s^n + a_{n-1}s^{n-1} + \ldots\ldots + a_1 s^1 + a_0}$$

توابع نقل شهيرة :

- مرشح تمرير منخفض

$$H_{LP}(s) = \frac{K_1}{s^2+Bs+w_o^2}$$

- مرشح تمرير مرتفع

$$H_{HP}(s) = \frac{K_2 s^2}{s^2+Bs+w_o^2}$$

- مرشح تمرير مجال

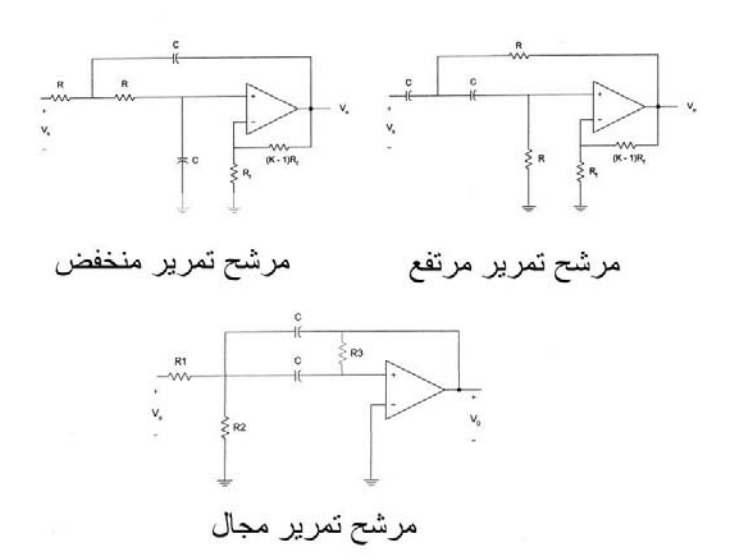
$$H_{BP}(s) = \frac{K_3 s}{s^2+Bs+w_o^2}$$

- مرشح منع مجال

$$H_{BR}(s) = \frac{K_4 s^2 + K_5}{s^2 + Bs + w_0^2}$$

حيث :

ثوابت k_1, k_2, k_3, k_4, B, and w_0



الاستجابة الترددية هي استجابة الشبكة لإشارة الدخل الجيبية ، فإذا استبدلنا ***s=jw*** في تابع الشبكة نحصل على :

$$H(s)|_{s=jw} = M(w)\angle\theta(w)$$

حيث :

$$M(w) = |H(jw)| \qquad \& \qquad \theta(w) = \angle H(jw)$$

وبرسم تغيرات $M(w)$ بالنسبة للتردد نحصل على الاستجابة الترددية للطويلة وكذلك برسم تغيرات $\theta(w)$ بالنسبة للتردد نحصل على الاستجابة الترددية للطور.

ويمكننا الحصول على هذه الميزات باستخدام تابع شهير في الماتلاب هو : **freqs** والذي تعطى الصيغة العامة له بالشكل : *hs = freqs(num,den,range)*

حيث :

$$num = [b_m\, b_{m-1}\, \ldots\ldots\, b_1\, b_0]$$

den = [a_n a_{n-1} ……. a_1 a_0]

range : مجال التردد للحالة المدروسة

hs : الاستجابة الترددية بالصيغة العقدية

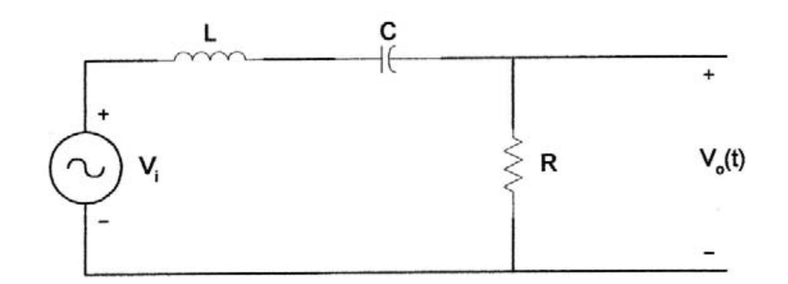
مثال :

من أجل الدارة المبينة بالشكل :

١- بين أن تابع النقل يعطى بالشكل : $H(s) = \frac{V_o(s)}{V_i(s)} = \frac{s\frac{R}{L}}{s^2+s\frac{R}{L}+\frac{1}{LC}}$

٢- إذا علمنا أن L = 5H, C =1.12µf, and R = 10000Ω بين شكل الاستجابة الترددية.

٣- إذا جعلنا قيمة المقاومة R = *100*Ω ما التغير الذي سيحدث ، بين بالرسم



$$H(s) = \frac{V_O(s)}{V_i(s)} = \frac{R}{R+sL+\frac{1}{sC}} = \frac{sCR}{s^2LC+sCR+1} \rightarrow H(s) = \frac{s\frac{R}{L}}{s^2+s\frac{R}{L}+\frac{1}{LC}}$$

والآن سنستعين بالماتلاب من أجل الرسم :

```
l=5;      c=1.25e-6;          r1=10000;            r2=100;
num1=[r1/l 0];
den1=[1 r1/l 1/(l*c)];
w=logspace(1,4);
h1=freqs(num1,den1,w);
f=w/(2*pi);
mag1=abs(h1);
phase1=angle(h1)*180/pi;
```

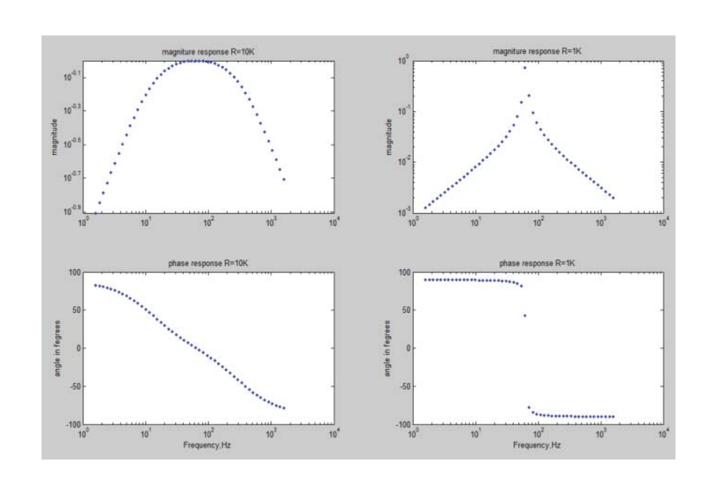
```
num2=[r2/l 0];
den2=[1 r2/l 1/(l*c)];
h2=freqs(num2,den2,w)
mag2=abs(h2);
phase2=angle(h2)*180/pi;

subplot(2,2,1)
loglog(f,mag1,'.')
title('magniture response R=10K')
ylabel('magnitude')

subplot(2,2,2)
loglog(f,mag2,'.')
title('magniture response R=1K')
ylabel('magnitude')

subplot(2,2,3)
semilogx(f,phase1,'.')
title('phase response R=10K')
xlabel('Frequency,Hz')
ylabel('angle in fegrees')

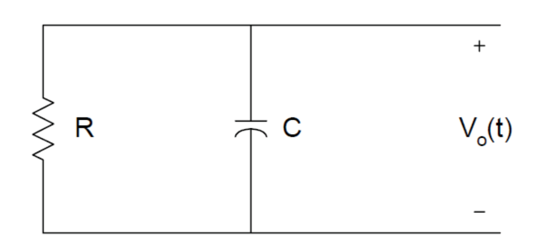
subplot(2,2,4)
semilogx(f,phase2,'.')
title('phase response R=1K')
xlabel('Frequency,Hz')
ylabel('angle in fegrees')
```



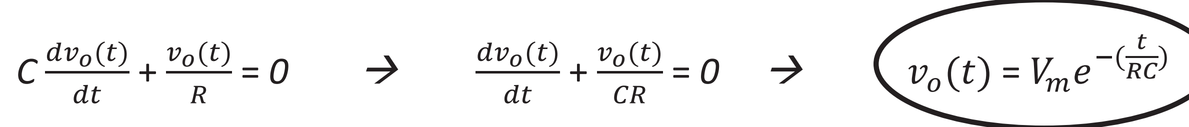
٣,١١ : دراسة الحالات العابرة فى الدارات الكهربائية

١,٣,١١ دراسة الحالة العابرة في دارة RC

١- الجهد عند تفريغ المكثف



حسب قانون كيرشوف :

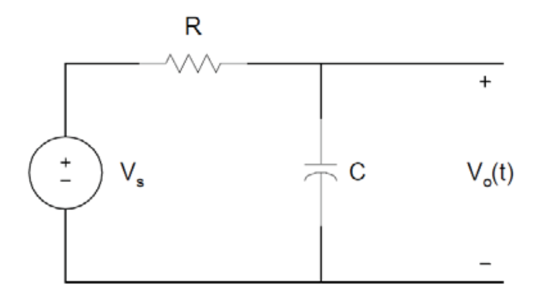


$$C\frac{dv_o(t)}{dt} + \frac{v_o(t)}{R} = 0 \quad \rightarrow \quad \frac{dv_o(t)}{dt} + \frac{v_o(t)}{CR} = 0 \quad \rightarrow \quad v_o(t) = V_m e^{-(\frac{t}{RC})}$$

حيث : τ = RC الثابت الزمني

المعادلة الأخيرة تمثل معادلة الجهد عند تفريغ المكثف .

٢- الجهد عند شحن المكثف



حسب قانون كيرشوف :

$$C\frac{dv_o(t)}{dt} + \frac{v_o(t) - V_s}{R} = 0 \quad \rightarrow \quad v_o(t) = V_s\,[1 - e^{-(\frac{1}{RC})}]$$

مثال (١) :

في حالة الشحن للمكثف ، لنفرض أن $C = 10\mu F$

اكتب برنامج لرسم الجهد بر المكثف في الحالات التاليتين :

- R = 1 KΩ

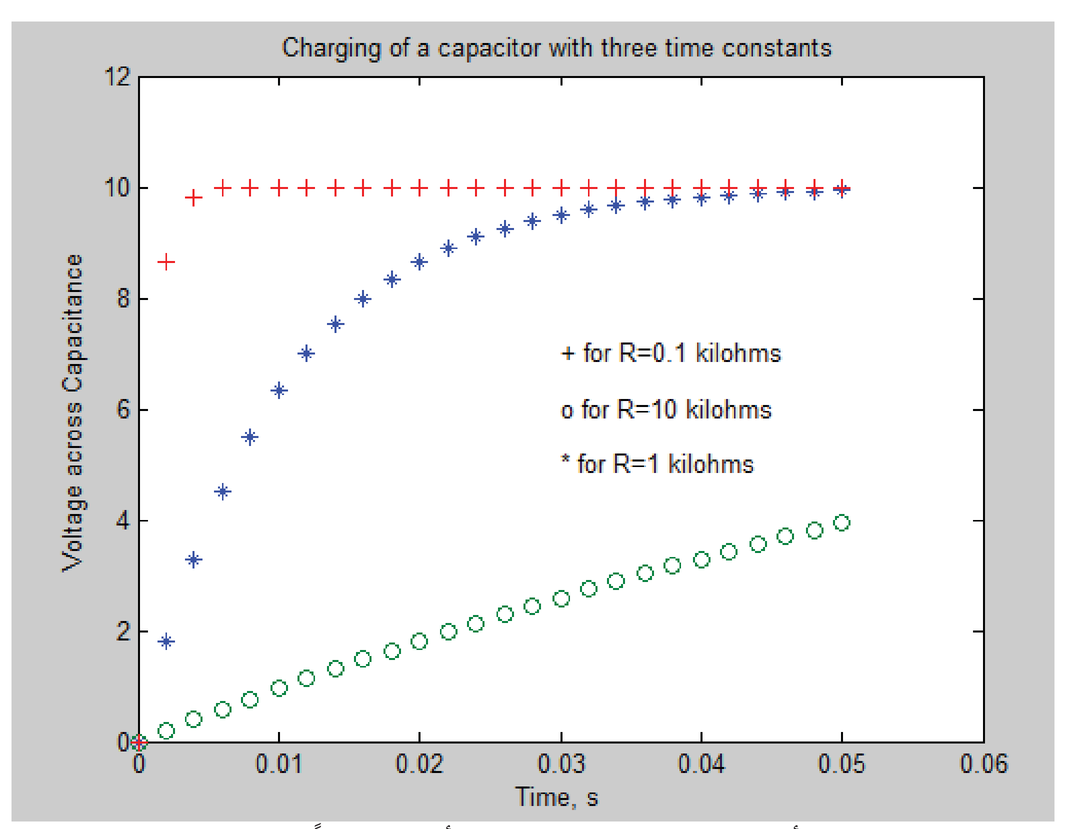
- R = 10 KΩ
- R = 0.1 KΩ

الحل :

Matlab Script

```
c=10e-6;
r1=1e3;
tau1=c*r1;
t=0:0.002:0.05;
v1=10*(1-exp(-t/tau1));
r2=10e3;
tau2=c*r2;
v2=10*(1-exp(-t/tau2));
r3=.1e3;
tau3=c*r3;
v3=10*(1-exp(-t/tau3));
plot(t,v1,'*',t,v2,'o',t,v3,'+')
axis([0 0.06 0 12])
title('Charging of a capacitor with three time constants')
xlabel('Time, s')
ylabel('Voltage across Capacitance')
text(0.03,5, '* for R=1 kilohms')
text(0.03,6, 'o for R=10 kilohms')
text(0.03,7, '+ for R=0.1 kilohms')
```

والشكل التالي يبين النتائج التي حصلنا عليها من البرنامج من أجل قيم مختلفة للمقاومة ..... وهو يمثل منحني شكل المكثف .



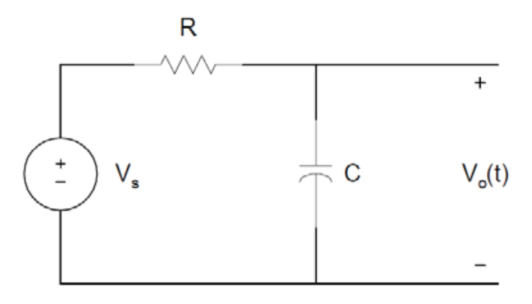
من الشكل الناتج نستنتج أنه كلما كانت الثابتة الزمنية أكثر صغراً كلما كان زمن شحن المكثف اصغر.

مثال (٢) :

في الدارة المبينة ، جهد الدخل هو نبضي بإشارة مربعة لها المطال 5V وعرض النبضة 0.5s، لو كانت قيمة $C = 10\,\mu F$ ، ارسم جهد الخرج v_o من أجل الحالات التالية :

- $R = 1000\ \Omega$
- $R = 10.000\ \Omega$

الرسم يبدأ من الصفر لينتهي زمن ثانية ونصف.



الحل :

Matlab Function

```
function [v,t]=rceval(r,c)
tau=r*c;
for i=1:50
    t(i)=i/100;
    v(i)=5*(1-exp(-t(i)/tau));
end
vmax=v(50);

for i=51:100
    t(i)=i/100;
    v(i)=vmax*exp(-t(i-50)/tau);
end
end
```

تذكرة :

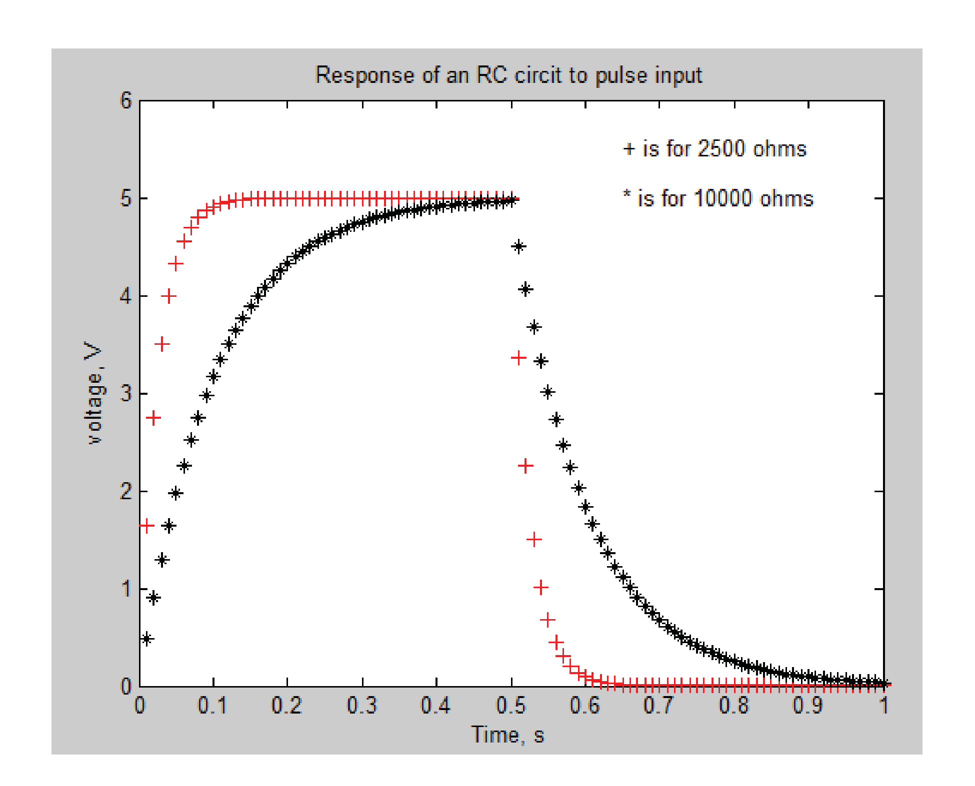
يجب تسمية ملف M-file الذي يحوي تابع بنفس اسم التابع لكي يعمل .

Matlab Script

```
c=10.0e-6;
r1=2500;
[v1,t1] = rceval(r1,c);
r2=10000;
[v2,t2] = rceval(r2,c);

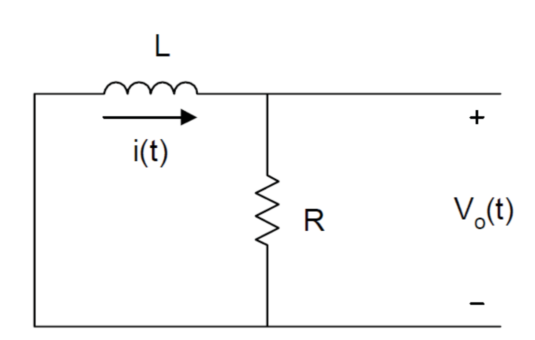
plot(t1,v1,'+r',t2,v2,'*k')
axis([0 1 0 6])
title('Response of an RC circit to pulse input')
xlabel('Time, s')
ylabel('voltage, V')
text(0.65,5.5,'+ is for 2500 ohms')
text(0.65,5.0,'* is for 10000 ohms')
```

والشكل الناتج (المبين بالشكل)يبين منحني الشحن والتفريغ للمكثف .



٣,٣,١١ دراسة الحالة العابرة في دارة RL

-١



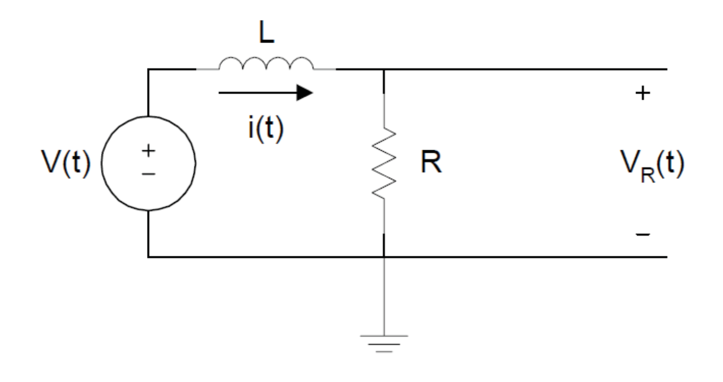
حسب قانون كيرشوف :

$$L\frac{di(t)}{dt} + R.i(t) = 0 \quad \rightarrow \quad i(t) = I_m.e^{-(\frac{t}{\tau})}$$

حيث : $\tau = \frac{L}{R}$ الثابت الزمني

إن المعادلة الأخيرة تمثل الاستجابة الطبيعية لدارة RL.

٢-



حسب قانون كيرشوف :

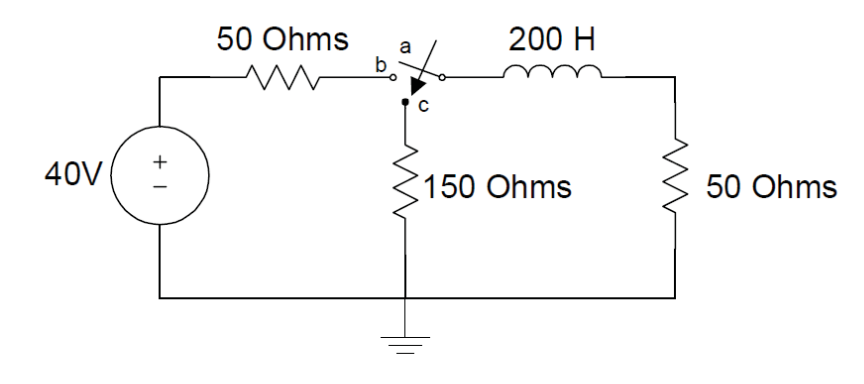
$$L\,\frac{di(t)}{dt} + R.i(t) = V_s \quad \rightarrow \quad i(t) = \frac{V_s}{R}\,(1 - e^{-(\frac{Rt}{L})})$$

الجهد عبر المقاومة :
$$V_R(t) = R.i(t) = V_s(1 - e^{-(\frac{Rt}{L})})$$

الجهد عبر المحارضة :
$$V_L(t) = V_s - V_R(t) = V_s.e^{-(\frac{Rt}{L})}$$

مثال :

من أجل الدارة المبينة بالشكل ، التيار المار في المحارضة معدوم ، في اللحظة t=0 القاطع ينتقل من النقطة a إلى النقطة b ويبقى لمدة ثانية واحدة ، وبعد ذلك ينتقل القاطع من النقطة b إلى النقطة c حيث يبقى لمدة غير محدودة ، بين شكل موجة التيار عبر المحارضة بالنسبة للزمن.



الحل :

- من أجل $0 < t < 1$ s ، نستخدم المعادلة :

$$i(t) = \frac{V_S}{R}\,(1 - e^{-(\frac{Rt}{L})}) = 0.4\,(1 - e^{-(\frac{t}{\tau 1})})$$

حيث : $\tau_1 = {}^{L}/_{R} = 200/100 = 2$ s

- عند اللحظة $t = 1$ s :

$$i(t) = 0.4\,(1 - e^{-0.5}) = I_{max}$$

- من أجل $t > 1$ s :

$$i(t) = I_{max}\, e^{-(\frac{t-0.5}{\tau_2})} \qquad \tau_2 = 200/200 = 1 \text{ s}$$

Matlab Script

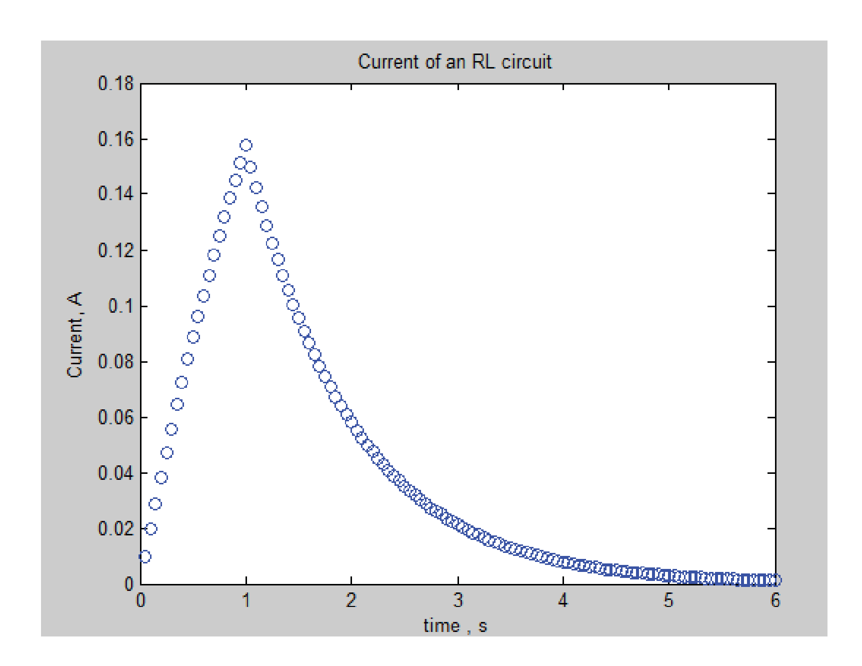
```
tau1 = 200/100;

for k=1:20
    t(k)=k/20;
    i(k)=0.4*(1-exp(-t(k)/tau1));
end

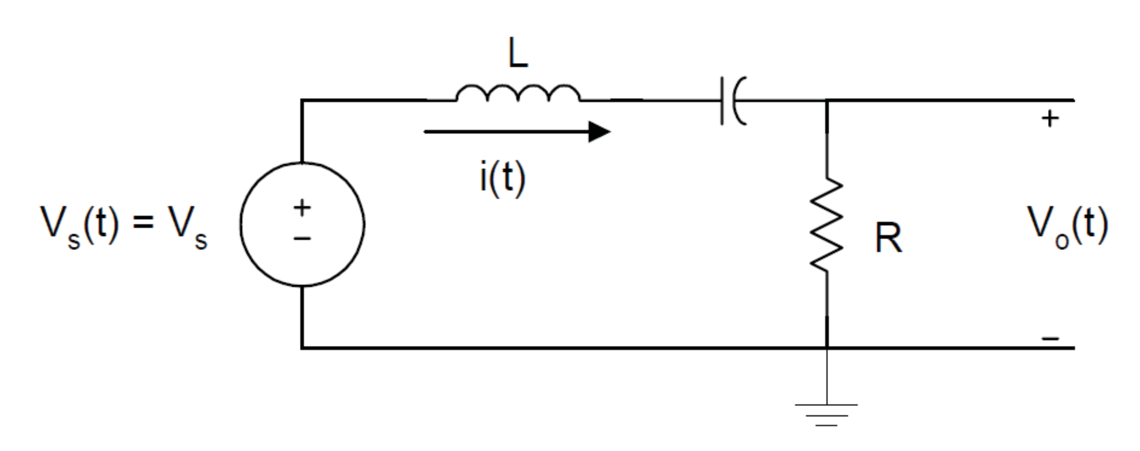
imax=i(20);
tau2=200/200;

for k=21:120
    t(k)=k/20;
    i(k)=imax*exp(-t(k-20)/tau2);
end

plot(t,i,'o')
axis([0 6 0 0.18])
title('Current of an RL circuit')
xlabel('time , s')
ylabel('Current, A')
```



٤,٣,١١ دراسة الحالة العابرة في دارة RLC



من أجل الدارة المبينة بالشكل ، وباستخدام قانون كيرشوف نكتب :

$$v_s(t) = L\,\frac{di(t)}{dt} + \frac{1}{C}\int_{-\infty}^{t} i(\tau)d\tau + Ri(t)$$

بمفاضلة المعادلة المبينة نحصل على المعادلة :

$$\frac{dv_s(t)}{dt} = L\frac{d^2i(t)}{dt^2} + R\frac{di(t)}{dt} + \frac{i(t)}{C}$$

يمكن الحصول على الحل المتجانس لهذه المعادلة بجعل $v_s(t) = constant$ وبالتالي :

$$0 = \frac{d^2i(t)}{dt^2} + \frac{R}{L}\frac{di(t)}{dt} + \frac{i(t)}{LC}$$

ويمكن كتابة هذه المعادلة بالشكل : $\beta^2 + a\beta + b = 0$

حيث : $a = R/L \qquad b = 1/LC$

وجذور المعادلة يمكن الحصول عليهم بسهولة وبفرض الجذور $\beta = \gamma , \vartheta$

فإن حل المعادلة السابقة يكون :

$$i_h(t) = A_1 e^{\alpha_1 t} + A_2 e^{\alpha_2 t} + A_3$$

مثال :

للدارة السابقة لنفرض أن :

L = 10 H , R = 400 Ohms and C = 100μF

$v_s(t)$ = 0 $\qquad i(0)$ = 4 A and $\frac{di(0)}{dt}$ = 15 A/s

والمطلوب إيجاد التيار $i(t)$.

الحل :

بجعل $v_s(t) = 0$ نكتب المعادلة :

$$0 = \frac{d^2 i(t)}{dt^2} + \frac{400}{10}\frac{di(t)}{dt} + 1000 i(t)$$

$$\beta^2 + 40\beta + 1000 = 0$$

وباستخدام الماتلاب يمكن إيجاد جذور هذه المعادلة :

Matlab Script

```
A=[1 40 1000];
root_A=roots(A)

    root_A =
            -20.0000 +24.4949i
            -20.0000 -24.4949i
```

وبعد أن حصلنا على الجذور يمكن كتابة علاقة التيار كما يلي :

$$i(t) = e^{-20t}(A_1\cos(24.4949t) + A_2\sin(24.4949t))$$

$$i(0) = e^{-0}(A_1 + A_2(0)) \rightarrow A_1 = 4$$

$$\frac{di(0)}{dt} = 24.4949A_2 - 20A_1 = 15 \rightarrow A_2 = 3.8$$

وبالنتيجة :

$$i(t) = e^{-20t}(4\cos(24.4949t) + 3.8\sin(24.4949t))$$

يمكن الاستعانة بتحويلات لابلاس من أجل الحصول على التوترات والتيارات في الدارات الأكثر تعقيداً و الجدول التالي يبين تحويل لابلاس لمجموعة من التوابع.

Laplace Transform Pairs

	f (t)	f(s)	
1	1	$\frac{1}{s}$	$s>0$
2	t	$\frac{1}{s^2}$	$s>0$
3	t^n	$\frac{n!}{s^{n+1}}$	$s>0$
4	e^{-at}	$\frac{1}{s+a}$	$s>a$
5	te^{-at}	$\frac{1}{(s+a)^2}$	$s>a$
6	$\sin(wt)$	$\frac{w}{s^2+w^2}$	$s>0$
7	$\cos(wt)$	$\frac{s}{s^2+w^2}$	$s>0$
8	$e^{at}\sin(wt)$	$\frac{w}{(s+a)^2+w^2}$	
9	$e^{at}\cos(wt)$	$\frac{s+a}{(s+a)^2+w^2}$	
10	$\frac{df}{dt}$	$sF(s)-f(0^+)$	
11	$\int_{0^+}^{t} f(\tau)d\tau$	$\frac{F(s)}{s}$	
12	$f(t-t_1)$	$e^{-t_1 s}F(s)$	

٤,١١ : نظرية الحالات المتغيرة State Variable technique

كطريقة ثانية لدراسة الحالة العابرة في الدارات الكهربائية الاستعانة بنظرية الحالات المتغيرة ، وهذه النظرية :

- يمكن استخدامها لتحليل نظم التحكم المختلفة.
- يمكن تطبيقها من أجل زمن متغير وللأنظمة غير الخطية.
- ملائمة للحلول الرقمية.
- يمكن استخدامها من أجل تطوير سلوك النظم المختلفة.

حيث تعتمد هذه النظرية على اختيار قيمة ما متغيرة أصغرية ومعروفة ولتكن t ، ومن ثم اختيار قيم أكبر من t لدراسة خرج النظام عندها.

وبشكل عام : لو فرضنا أن x متغير الحالة ، u هو دخل النظام ، y هو خرج النظام المدروس فيمكننا أن نكتب :

$$x\dot{(t)} = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

حيث :

$$u(t) = \begin{matrix} u_1(t) \\ u_2(t) \\ . \\ . \\ u_n(t) \end{matrix} \qquad x(t) = \begin{matrix} x_1(t) \\ x_2(t) \\ . \\ . \\ x_n(t) \end{matrix} \qquad y(t) = \begin{matrix} y_1(t) \\ y_2(t) \\ . \\ . \\ y_n(t) \end{matrix}$$

وكلاً من : $A, B, C \; \& \; D$ مصفوفات تعين من ثوابت النظام.

مثال :

ليكن نظام التحكم المعبر عنه بالمعادلة التفاضلية التالية (دخل واحد و خرج واحد) ....

$$\frac{d^4y(t)}{dt^4} + 3\frac{d^3y(t)}{dt^3} + 4\frac{d^2y(t)}{dt^2} + 8\frac{dy}{dt} + 2y(t) = 6u(t)$$

الحل :

$$x_1(t) = y(t)$$
$$x_2(t) = \frac{dy}{dt} = \frac{dx_1(t)}{dt} = x_1\dot{(t)}$$
$$x_3(t) = \frac{d^2y}{dt^2} = \frac{dx_2(t)}{dt} = x_2\dot{(t)}$$

$$x_4(t) = \frac{d^3y}{dt^3} = \frac{dx_3(t)}{dt} = \dot{x_3}(t)$$

$$x_5(t) = \frac{d^4y}{dt^4} = \frac{dx_4(t)}{dt} = \dot{x_4}(t)$$

بتعويض المعادلات الأخيرة بالمعادلة المعطاة في نص السؤال نجد :

$$\dot{x_4}(t) = 6u(t) - 3x_4(t) - 4x_3(t) - 8x_2(t) - 2x_1(t)$$

وبالتالي نكتب :

$$\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \\ \dot{x_3}(t) \\ \dot{x_4}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -2 & -8 & -4 & -3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 6 \end{bmatrix} u(t)$$

ويمكن التعبير عن المصفوفة بالمعادلة :

$$\dot{x}(t) = Ax(t) + Bu(t)$$

حيث :

$$\dot{x} = \begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \\ \dot{x_3}(t) \\ \dot{x_4}(t) \end{bmatrix} \qquad A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -2 & -8 & -4 & -3 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 6 \end{bmatrix}$$

أما معادلة خرج النظام :

$$y(t) = Cx(t) + Du(t)$$

وبما أن :

$$x_1(t) = y(t)$$

فإن :

$$C = [1 \quad 0 \quad 0 \quad 0]$$

$$D = [\ 0\ ]$$

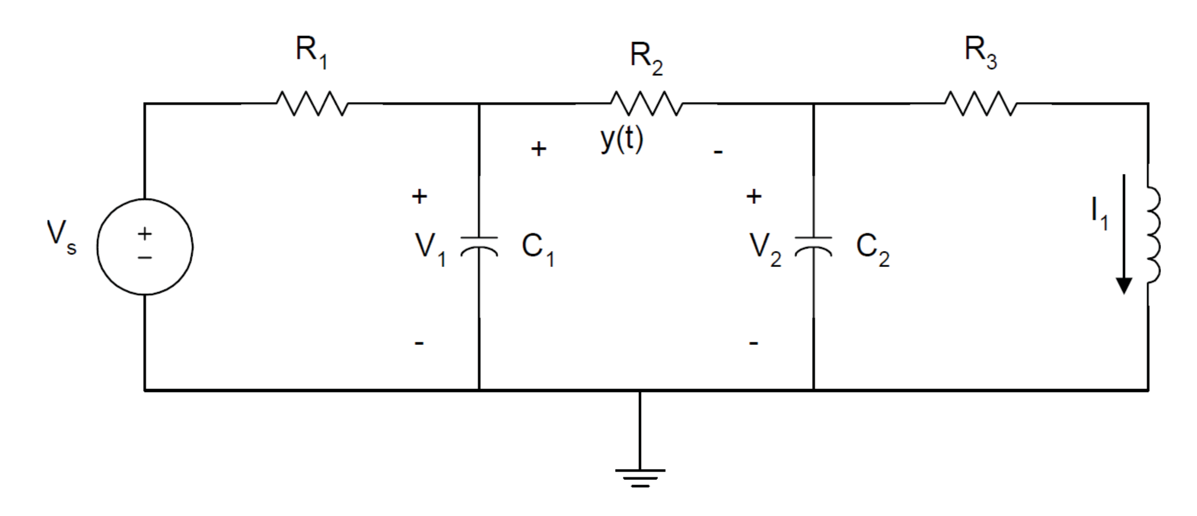
يمكن اختيار المتغيرات في الدارات الكهربائية بالاستعانة بالأفكار التالية :

- التيارات المارة عبر الملفات (الحثية أو المحارضة) تكون دوماً متغيرة.
- التوترات على المكثفات تكون دوماً متغيرة.
- التوترات والتيارات المارة في المقاومات تكون دوماً ثابتة القيمة.

- في الحلقات المغلقة التي تحوي مكثفات أو نقاط الاتصال التي تحوي ملفات يتم اختيار المتغيرات بالاعتماد على أول قاعدتين.

مثال :

للدارة المبينة بالشكل :



بالاعتماد على القواعد الأخيرة تم اختيار المتغيرات لتكون $V_1, V_2 and\ i_1$.

والمطلوب استنتاج المصفوفة التي تصف النظام في الحالة العابرة.

الحل :

بتحليل العقد يمكن أن نكتب :

$$C_1 \frac{dv_1(t)}{dt} + \frac{V_1 - V_S}{R_1} + \frac{V_1 - V_2}{R_2} = 0$$

$$C_2 \frac{dv_2(t)}{dt} + \frac{V_2 - V_1}{R_2} + i_1 = 0$$

بتحليل الحلقات يمكن أن نكتب :

$$V_2 = i_1 R_3 + L \frac{di_1(t)}{dt}$$

وتعطى علاقة الخرج :

$$y(t) = v_1(t) - v_2(t)$$

وبتبسيط المعادلات الأخيرة :

$$\frac{dv_1(t)}{dt} = -(\frac{1}{C_1R_1} + \frac{1}{C_1R_2})V_1 + \frac{V_2}{C_1R_2} + \frac{V_s}{C_1R_1}$$

$$\frac{dv_2(t)}{dt} = \frac{1}{C_2R_2}V_1 - \frac{V_2}{C_2R_2} - \frac{i_1}{C_2}$$

$$\frac{di_1(t)}{dt} = \frac{V_2}{L} - \frac{R_3i_1}{L}$$

من المعادلات الثلاثة الأخيرة نشكل المصفوفة :

$$\begin{bmatrix} \dot{V_1} \\ \dot{V_2} \\ \dot{i_1} \end{bmatrix} = \begin{bmatrix} -(\frac{1}{C_1R_1} + \frac{1}{C_1R_2}) & \frac{1}{C_1R_2} & 0 \\ \frac{1}{C_2R_2} & -\frac{1}{C_2R_2} & -\frac{1}{C_2} \\ 0 & \frac{1}{L} & -\frac{R_3}{L} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ i_1 \end{bmatrix} + \begin{bmatrix} \frac{1}{C_1R_1} \\ 0 \\ 0 \end{bmatrix} V_s$$

وتكون علاقة الخرج :

$$y = \begin{bmatrix} 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ i_1 \end{bmatrix}$$

التوابع ODE المتاحة في الماتلاب تتيح لنا إمكانية حل المعادلات التفاضلية و هذا ما سنبينه بالفقرة التالية.

التابع ode وكيفية استخدامه :

هناك في الماتلاب تابعين : ode23 & ode45 لإيجاد الحلول العددية للمعادلات التفاضلية الاعتيادية.

- التابع ode23 : يعتمد هذا التابع في مكاملة المعادلات التفاضلية على صيغة رانج غوتا (Runge-Kutta) من المرتبة الثانية و الثالثة.
  و الصيغة العامة للتابع :

```
[t,x] = ode23(xprime , tstart , tfinal , xo , tol , trace)
```

- التابع ode45 : يعتمد هذا التابع في مكاملة المعادلات التفاضلية على صيغة رانج غوتا أيضاً ولكن من المرتبة الرابعة والخامسة.
  والصيغة العامة للتابع :

```
[t,x] = ode45(xprime , tstart , tfinal , xo , tol , trace)
```

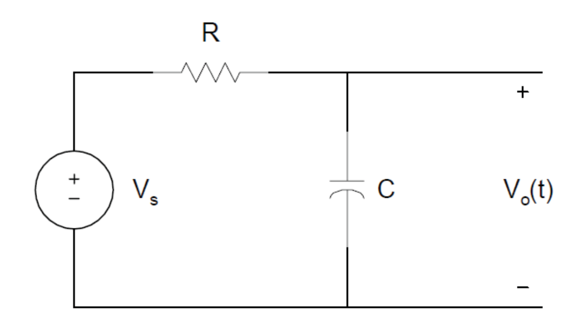
حيث :

- xprime : اسم التابع في برنامج الماتلاب الذي يحوي المعادلة التفاضلية ، حيث التابع سيحسب شعاع الحالة المشتق وذلك بإعطائه الزمن t وشعاع الحالة $x(t)$ ليعطي بالنتيجة الخرج $xdot$.

- tstart : زمن البدء للتكامل.
- tfinal : زمن انتهاء التكامل.
- xo : شعاع الحالة الأولية.
- tol : خيارية ويمكن إهمالها ووظيفتها تحديد دقة الحل.

مثال(١) :

في الشكل المبين لنفرض أن : $C = 10\mu F$, R=10000Ω , V_S = 10V والمطلوب إيجاد توتر الخرج بين (0→20ms) ، افرض أن $v_o(0) = 0$ وذلك بالطريقتين السابقتين الذكر.



الحل :

نكتب المعادلة :

$$C\,\frac{dv_O(t)}{dt} + \frac{v_O(t) - V_S}{R} = 0$$

وبالتالي :

$$\frac{dv_O(t)}{dt} = \frac{V_S}{CR} - \frac{v_O(t)}{CR} = 100 - 10v_o(t)$$

ومن المعادلة التالية يمكن إيجاد الحل التحليلي :

$$v_o(t) = 10(1 - e^{-(\frac{t}{CR})})$$

Matlab Script

```
t0=0;
tf=20e-3;
xo=0;
[t,vo]=ode23('diff1',t0,tf,xo);

v0_analy=10*(1-exp(-10*t));

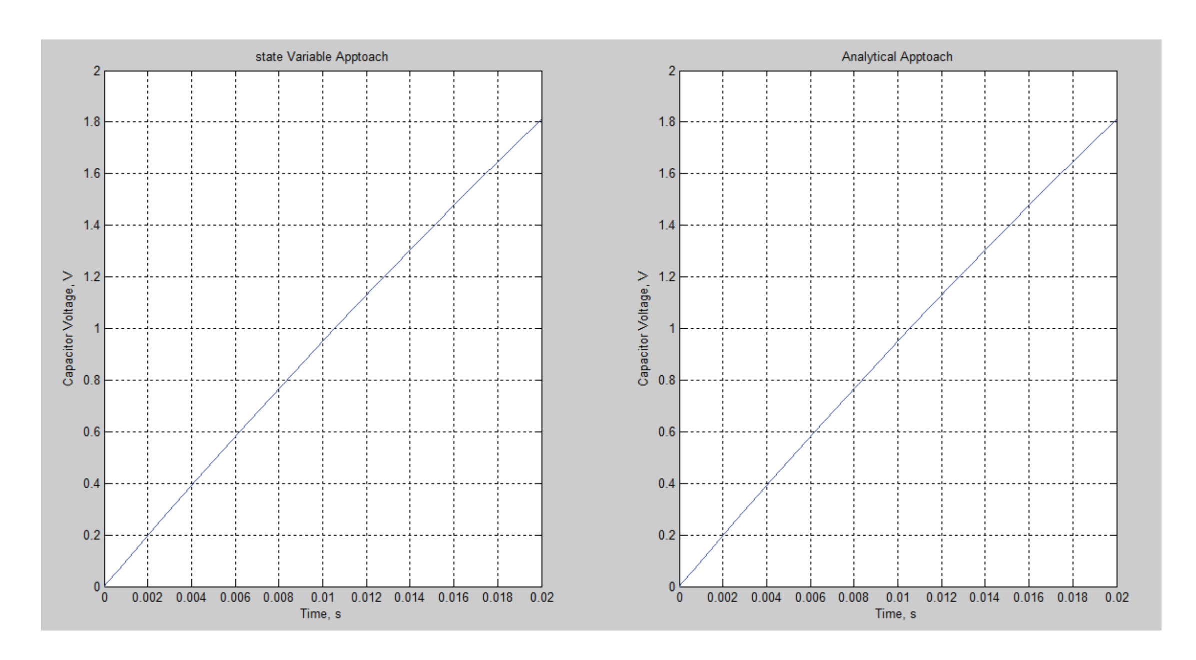
subplot(1,2,1)
```

```
plot(t,vo,'b')
title('state Variable Apptoach')
xlabel('Time, s')
ylabel('Capacitor Voltage, V')
grid
subplot(1,2,2)
plot(t,v0_analy,'b')
title('Analytical Apptoach')
xlabel('Time, s')
ylabel('Capacitor Voltage, V')
grid
```

Matlab Function

```
function dy = diff1(t,y)
dy = 100 - 10*y;
end
```

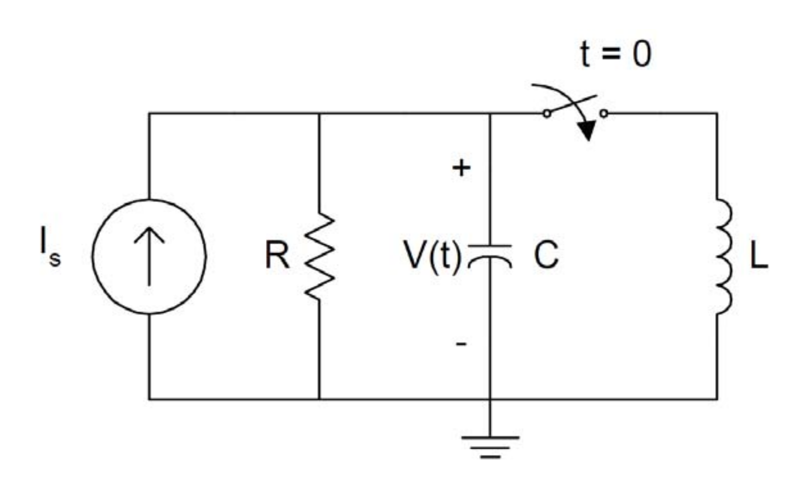
وبعد تنفيذ البرنامج حصلنا على النتائج التالية ....



مثال(٢) :

للدارة المبينة بالشكل .... لو فرضنا أن : R = 10Ω , L = 1/32 H , C = 50μF والمطلوب إيجاد قيمة التوتر $v(t)$ بالطريقتين والمقارنة بين النتائج.

$$v_c(0) = 20V \qquad i_L(0) = 0$$



الحل :

$$L\ \frac{di_L(t)}{dt} = v_c(t)$$

$$C\ \frac{dv_c(t)}{dt} + i_L + \frac{v_c(t)}{R} - I_s = 0$$

وبتبسيط المعادلات :

$$\frac{di_L(t)}{dt} = \frac{v_c(t)}{L}$$

$$\frac{dv_c(t)}{dt} = \frac{I_s}{C} - \frac{i_L}{C} - \frac{v_c(t)}{CR}$$

لنفرض أن :

$$x_1(t) = i_L(t)$$

$$x_2(t) = v_c(t)$$

فنحصل بالنتيجة على :

$$\dot{x_1}(t) = \frac{1}{L}x_2(t)$$

$$\dot{x_2}(t) = \frac{I_S}{C} - \frac{1}{C}x_1(t) - \frac{1}{RC}x_2(t)$$

ويمكن الآن كتابة ملف m-file بالماتلاب بالاستعانة بالمعادلات التفاضلية المبينة.

```
function xdot=diff2(t,x)

is=2;
c=50e-6;
L=1/32;
r=10;
k1=1/c;
k2=1/L;
k3=1/(r*c);

xdot(1)=k2*x(2);
xdot(2)=k1*is-k1*x(1)-k3*x(2);
end

t0=0;
tf=30e-3;
x0=[0 20];
[t,x]=ode23('diff2',t0,tf,x0);

subplot(2,1,1)
plot(t,x(:,2))
xlabel('Time, s')
ylabel('Capacitor voltage, V')
text(0.01,7,'state Variable Approach')

t2=0:1e-3:30e-3;
vt=-6.667*exp(-1600*t2)+26.667*exp(-400*t2);
subplot(2,1,2)
plot(t2,vt)
xlabel('Time, s')
ylabel('Capacitor voltage, V')
text(0.01,4.5,'Result from previous Example')
```

٥,١١ : رباعيات الأقطاب

سأكتفي بمثال لشرح رباعيات الأقطاب.

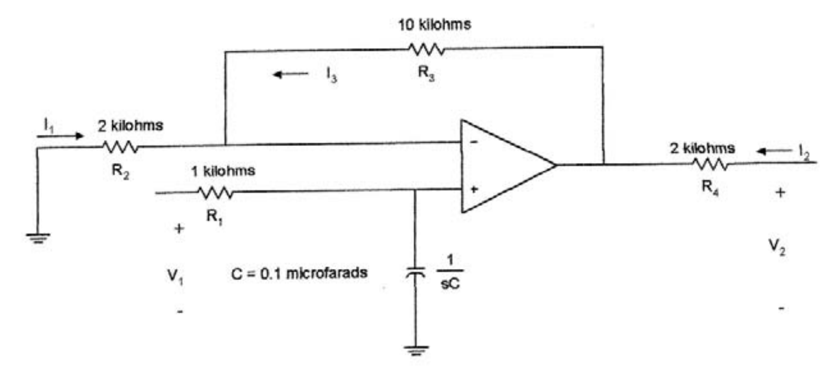
مثال :

بفرض إن مكبر العمليات الوضح بالشكل هو مثالي فالمطلوب :

١- أوجد العناصر الأساسية لمصفوفة الممانعات z.

٢- إذا كانت الشبكة موصلة بمنبع جهد بحيث مقاومته الداخلية 50Ω ومقاومة الحمل 1kΩ، أوجد نسبة التكبير.

٣- باستخدام الماتلاب ارسم الاستجابة الترددية للمطال.



باستخدام قانون كيرشوف نجد :

$$V_1 = R_1 I_1 + \frac{I_1}{sC}.........(1)$$

$$V_2 = R_4 I_2 + R_3 I_3 + R_2 I_3$$

$$R_2 I_3 = \frac{I_1}{sC}$$

من المعادلتين الأخيرتين :

$$V_2 = \frac{(R_2 + R_3) I_1}{sCR_2} + R_4 I_2.........(2)$$

من المعادلتين ١-٢ نجد :

$$z_{11} = R_1 + \frac{1}{sC}$$

$$z_{12} = 0$$

$$z_{21} = (1 + \frac{R_3}{R_2})(\frac{1}{sC})$$

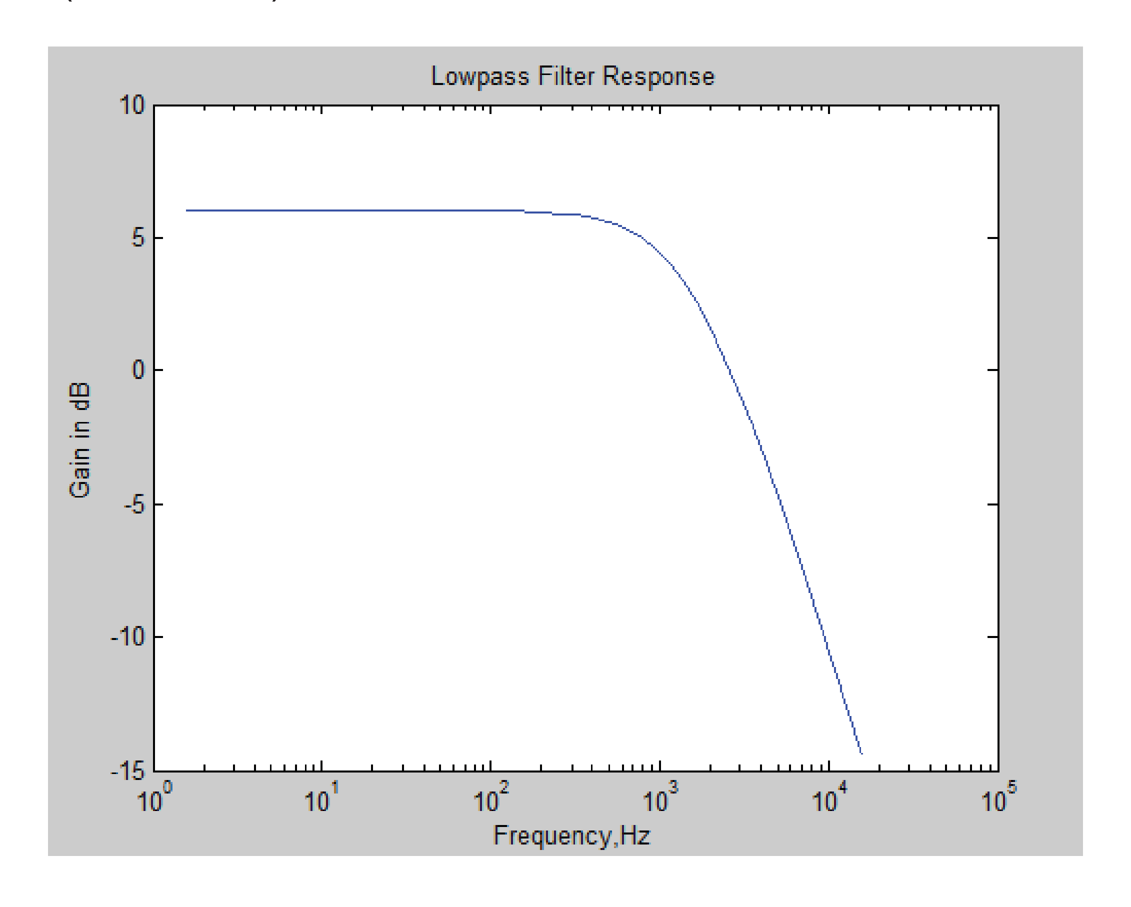
$$z_{22} = R_4$$

ولإيجاد نسبة التكبير :

$$\frac{V_2}{V_g} = \frac{z_{21}.Z_L}{(z_{11} + Z_g)(z_{22} + Z_L) - z_{12} z_{21}} = \frac{2}{[1 + 1.05 * 10^{-4} s]}$$

ومن أجل الرسم باستخدام الماتلاب نكتب البرنامج التالي :

```
clc
clear
num=[2];
den=[1.05e-4 1];
w=logspace(1,5);
h=freqs(num,den,w);
f=w/(2*pi);
mag=20*log10(abs(h));
semilogx(f,mag);
title('Lowpass Filter Response')
xlabel('Frequency,Hz')
ylabel('Gain in dB')
```





أساسيات المحاكاة في الماتلاب

- المكتبة Simulink.
- الإدخال و الإخراج و التعامل مع نماذج المحاكاة .
- تمثيل نظم التحكم .
- تمثيل و محاكاة الدارات الكهربائية .

أساسيات المحاكاة فى الماتلاب

١,١٢ مقدمة

في هذا الفصل سوف نعرض الخطوات الواجب اتباعها من أجل الحصول على العناصر المناسبة لتشكيل نموذج المحاكاة باستخدام بيئة الماتلاب بدءاً من فتح صفحة النموذج (Model) وحتى تنفيذ أمر النمذجة.

لقد وضعت مجموعة من الأشكال اللاحقة تبين خطوات الحصول على العناصر اللازمة لبناء الموديل (النموذج) بدءاً من المكتبة Simulink ، حيث تحتوي هذه المكتبة على مجموعة من المكتبات الفرعية وكل مكتبة فرعية تضم مجموعة من العناصر التي نستخدمها في بناء المخطط الصندوقي للمحاكاة.

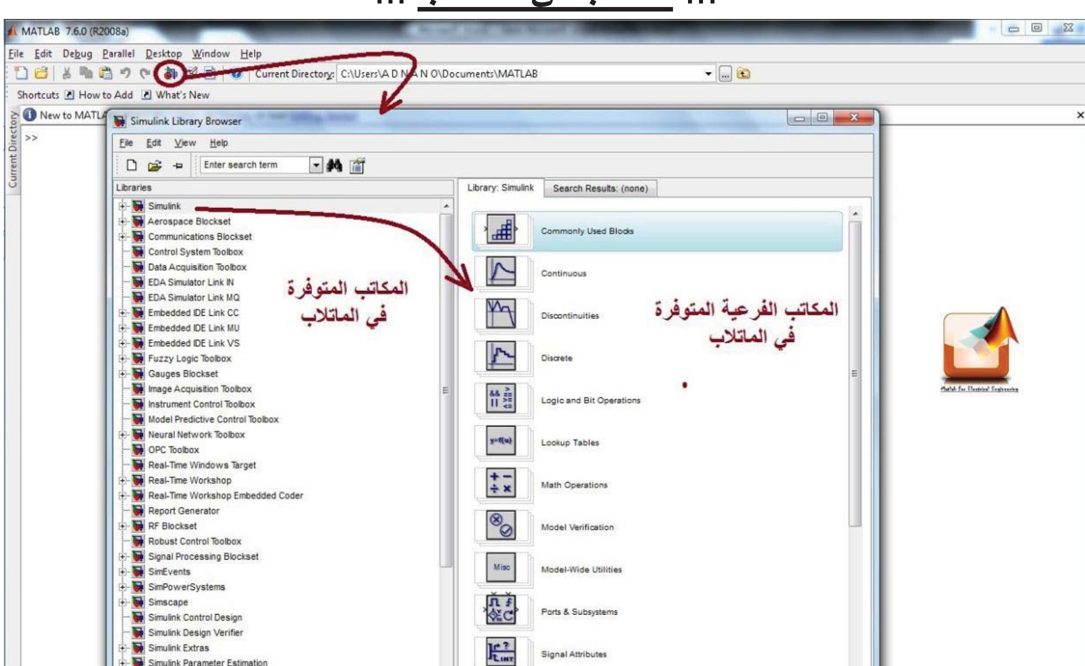
٢,١٢ إظهار محتويات بعض المكتبات

قبل البدء ببناء المخططات الصندوقية لابد من التعرف على المكتبات الفرعية وبعض العناصر الموجودة في هذه المكتبات تمهيداً للبدء بعملية بناء المخطط (الموديل) اللازم لعملية محاكاة بعض الدارات الكهربائية.

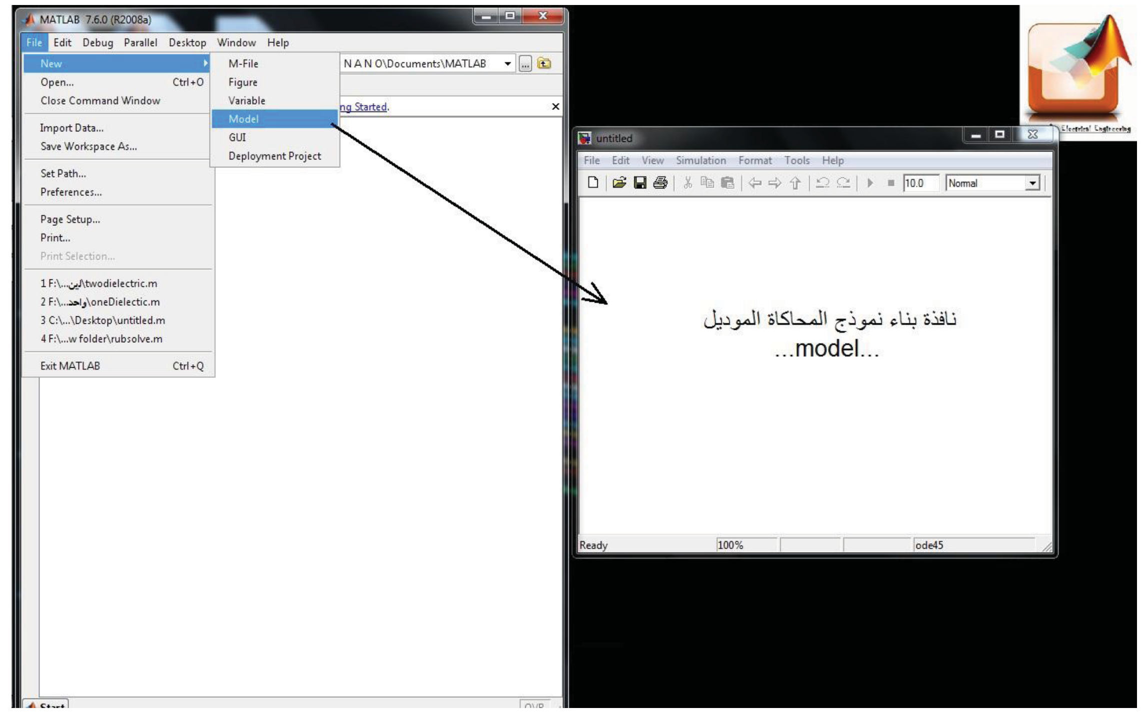
بعد ذلك يمكن الانتقال إلى بناء نماذج أكثر تعقيداً تمثل عناصر نظم القدرة الكهربائية مثل الآلات الكهربائية ، الشبكات ، القواطع ، عناصر إلكترونيات القدرة الكهربائية .... إلخ. بعد استنتاج النموذج الرياضي لهذه العناصر.

تبين الأشكال التالية كيفية الوصول إلى المكتبات الخاصة بالماتلاب ، كما تعرض لنا بعض العناصر الموجودة ضمن كل مكتبة فرعية ويمكن الاطلاع على جميع العناصر عند تشغيل البرنامج.

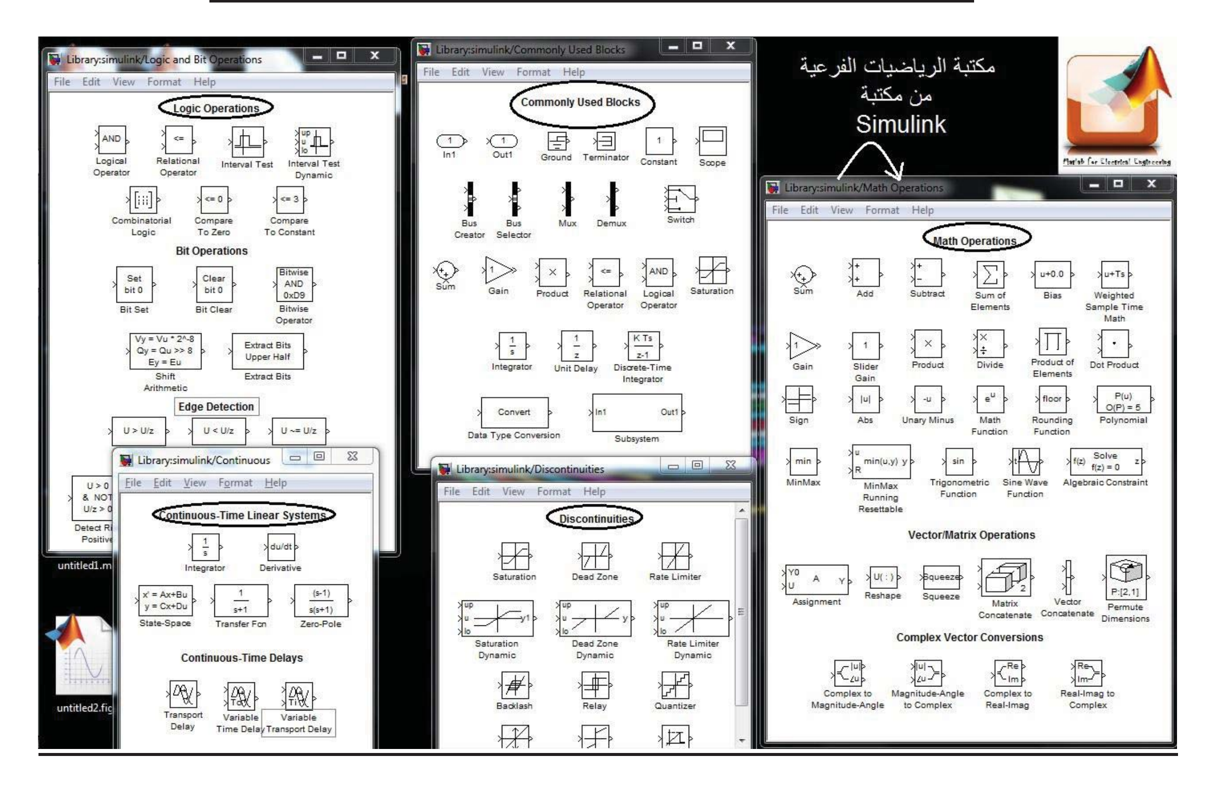
... المكاتب في الماتلاب ...



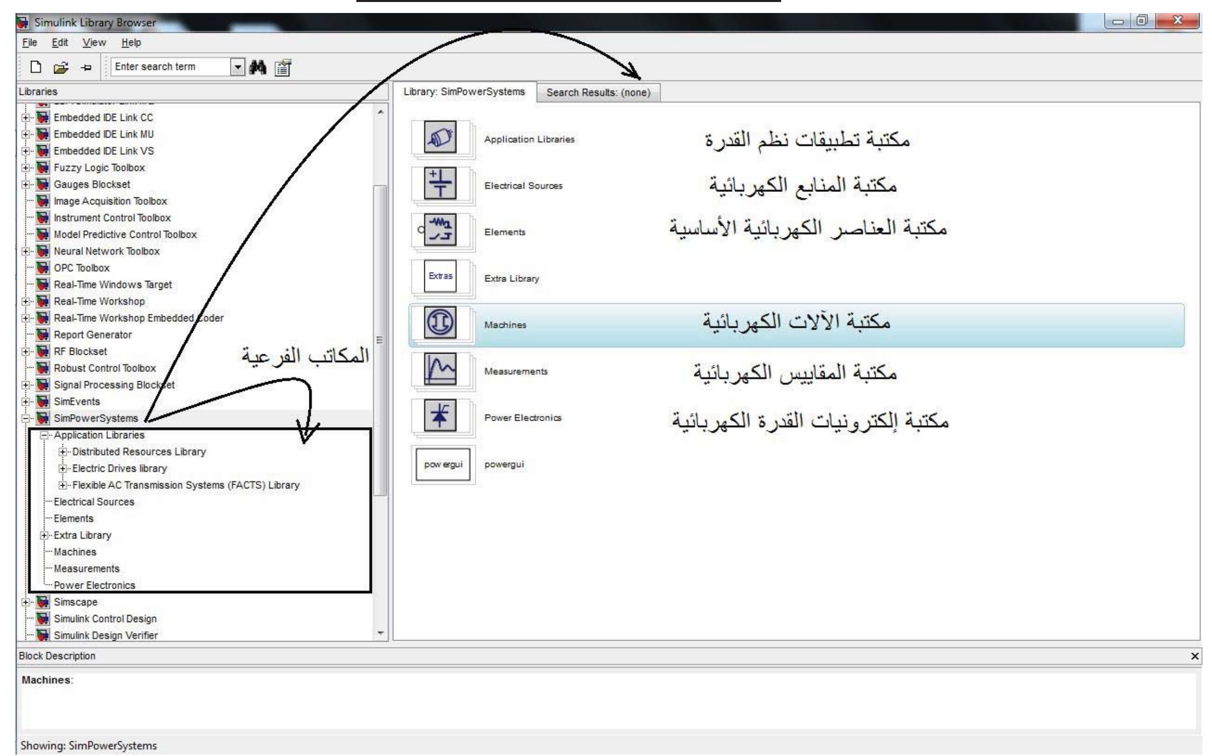
... خطوات فتح صفحة لرسم المخطط الصندوقي ...



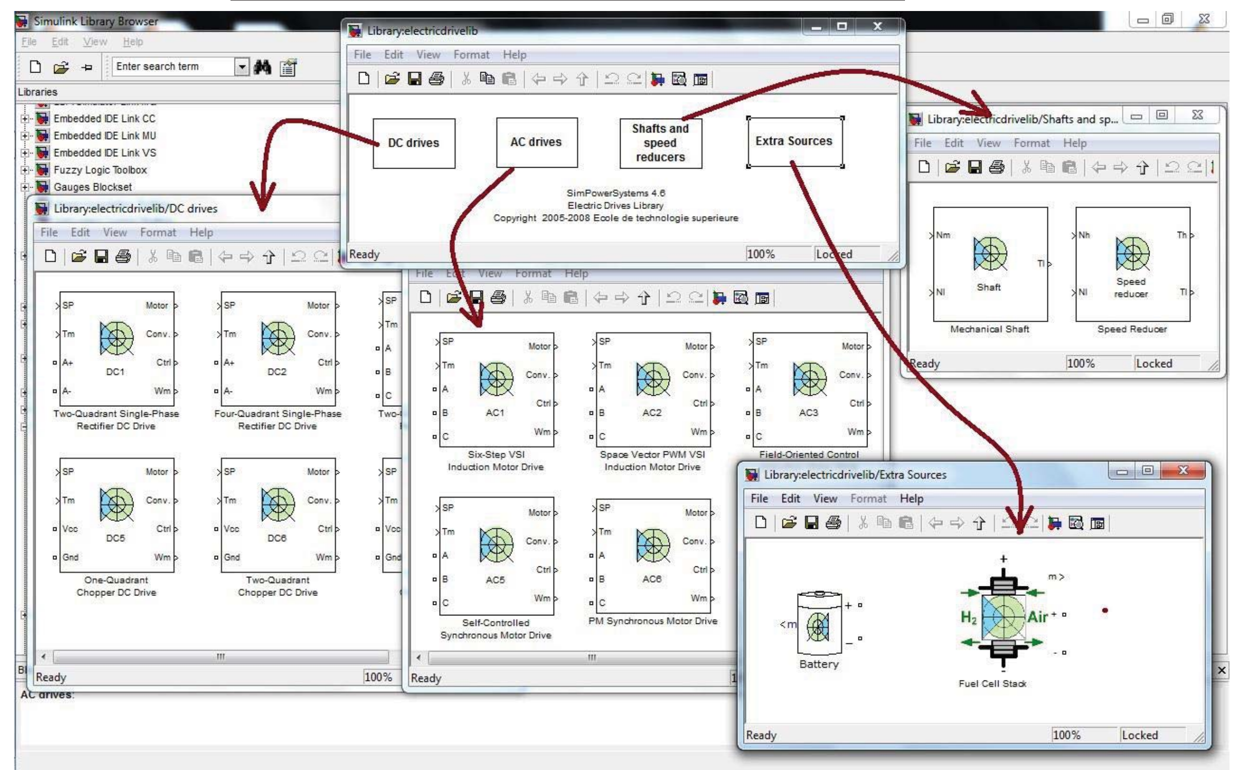
... محتويات بعض مكونات المكاتب الفرعية لمكتبة السيميولينك ...



…simpower systems library…



…Application Library of simpowersystems …



٣,١٢ تعريف المكتبة أو الأداة Simulink

هي عبارة عن منهج برمجي نستطيع من خلاله نمذجة ومحاكاة وتحليل الأنظمة الديناميكية سواء الخطية وبزمن مستمر أو متقطع ، ولتسهيل النمذجة توفر المكتبة Simulink إمكانية البرمجة البيانية الصندوقية باستخدام الفأرة وبناء التوابع المناسبة ، والوصول للبرمجة ومخطط يعبر عن المسألة المطروحة ، ويمكن أخذ تلك الصناديق من مكتبات Simulink الواسعة والشاملة لكل النماذج الممكنة ، سواء مكتبة المصادر أو مكتبة الخرج أو الكتل المعبرة عن الحالات الخطية و اللاخطية ، أو أدوات الوصل بين تلك الكتل. كما نستطيع صياغة وبناء الكتل الخاصة بنا ، ونستطيع الولوج إلى داخل الكتلة بالنقرعليها نقراً مزدوجاً فتصبح بمستوى برمجي أعمق ، وهكذا يمكننا التنقل ضمن مستويات مختلفة تعبر عن هيكلية المسألة المطروحة بشكل واضح.

بعد بناء المخطط الصندوقي والانتهاء من ذلك يمكن إجراء المحاكاة عن طريق التكامل وبطرائق مختلفة ، وكما ويمكن إظهار النتائج ومراقبتها في أثناء المكاملة باستخدام صناديق الخرج والتي تؤمن خيارات كثيرة وطرقاً عديدة لإخراج وإظهار النتائج أو إرسالها لملف أو تخزينها ، وسوف نتعرض على بعض الطرائق والإمكانيات لبناء المخططات الصندوقية اللازمة لمحاكاة بعض الدارات الكهربائية من خلال مجموعة من الأمثلة.

مثال (١) :

جمع عدد من التوابع الجيبية (sin) و دراسة تأثير التوافقيات على شكل الإشارة الجيبية الناتجة .

- نحتاج في هذا المثال إلى العناصر التالية :

١- منبع إشارة sinwave.

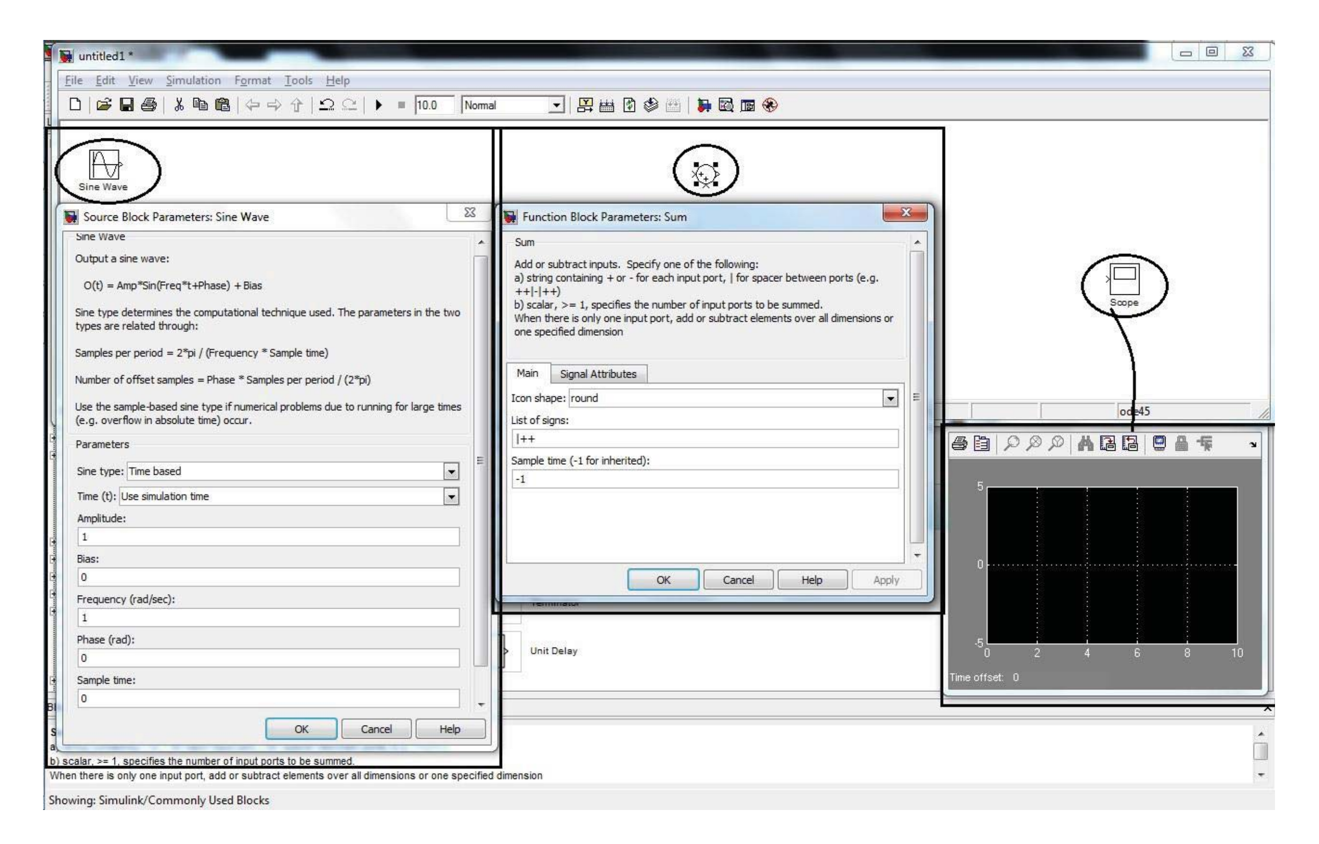
٢- عنصر جمع sum.

٣- عنصر إظهار لإظهار شكل الإشارة الناتجة Scope.

والشكل التالي يبين هذه العناصر وخصائص كل عنصر من هذه العناصر ، حيث نلاحظ أن إشارة الدخل الجيبية يتم التعامل معها بالعلاقة التالية :

$$Out = Amp*sin(Freq*t + Phase) + Bias$$

أما إشارة الجمع sum فيمكن زيادة عدد الإشارات الداخلة إليها من خلال التعديل في List of signs وذلك بإضافة إشارات الجمع (+) بعدد عدد الدخل المرادة.



ويمكن إضافة أي عنصر من العناصر السابقة إلى النموذج الجديد الذي نريد العمل فيه بالضغط على العنصر بالزر اليميني للماوس ومن ثم نختار (..... add to) ليفتح لنا صفحة (model) جديدة تحوي العنصر المضاف.



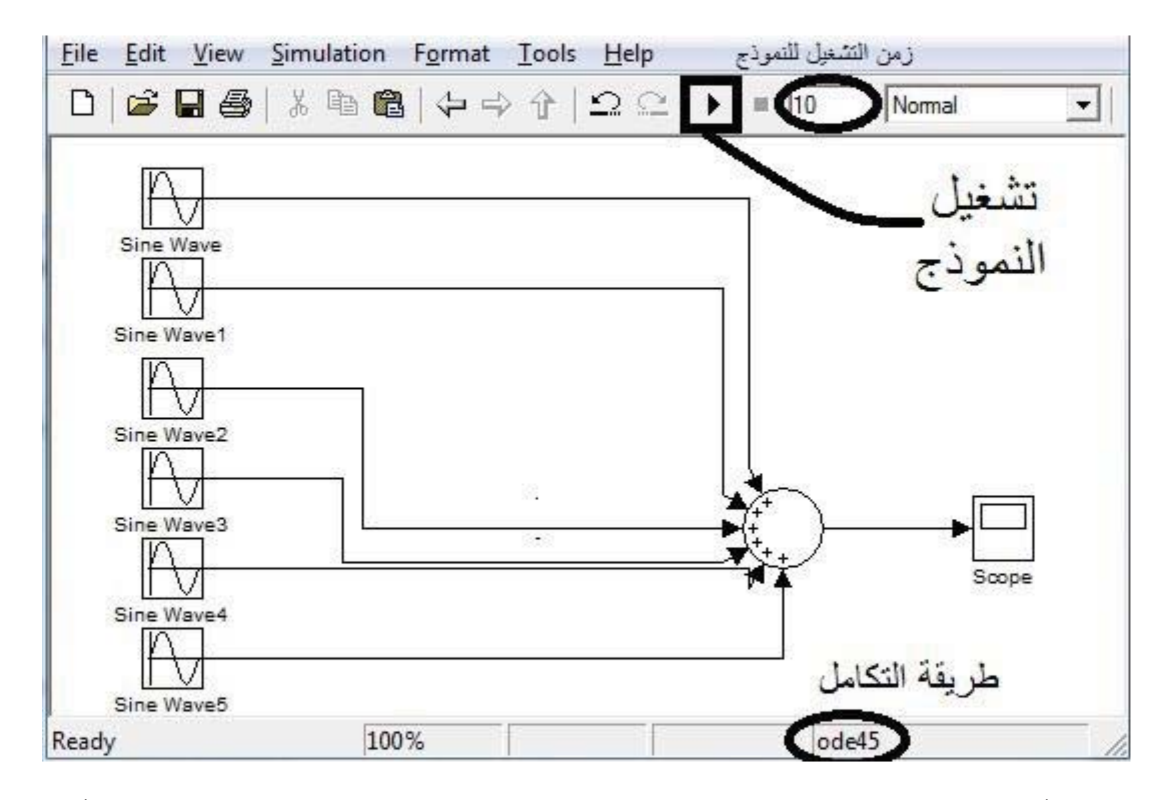
يمكن العثور على أي عنصر نريد إضافته بكتابة اسم العنصر في مربع البحث في أعلى الشاشة ، وبهذه الطريقة لا داعي للبحث طويلاً عن العنصر .

نقوم الآن بالتعديل على الإشارات الجيبية المراد جمعها وذلك كما يبين الجدول التالي :

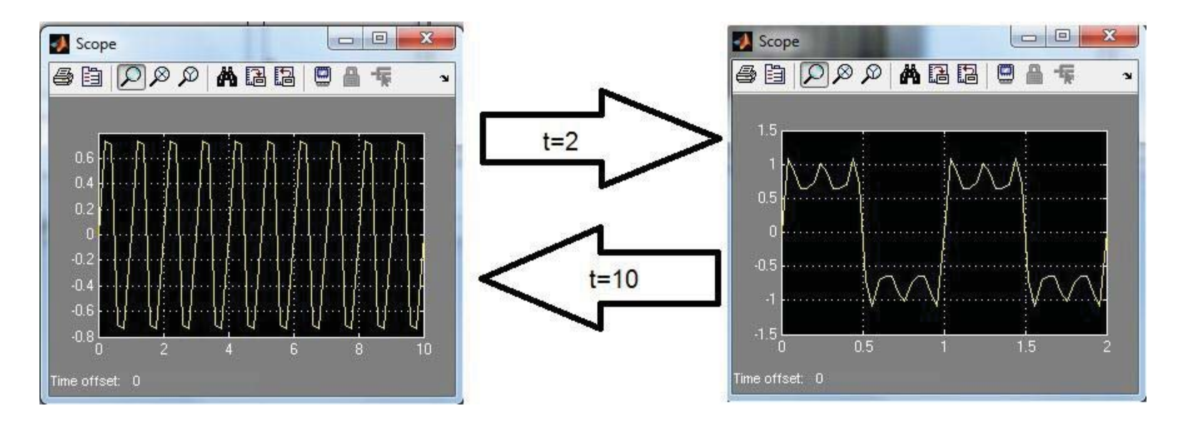
رقم الإشارة الجيبية	Amp.	Freq.	Bias
١	1	2π	0
٢	1/3	$3*2\pi$	0
٣	1/5	$5*2\pi$	0
٤	1/7	$7*2\pi$	0
٥	1/9	$9*2\pi$	0
٦	1/11	$11*2\pi$	0

بعد ذلك نقوم بالتعديل على عدد مداخل عنصر الجمع sum بحيث يكون عدد المداخل مساوي لعدد الإشارات الداخلة إليه وفي مثالنا هذا (6) وبالتالي نكتب في مربع الحوار List of signs العبارة التالية (++++++|) ونضغط موافق فنلاحظ تغير عدد المداخل.

بعد تنفيذ النموذج (model) وترتيب عناصره نلاحظ أن شكل المخطط يصبح كما يلي :



نلاحظ أننا لو قمنا بتشغيل النموذج وجعل الزمن كبير 10s كما هو مبين بالشكل فإن أثر التوافقيات لن يظهر ولذلك فإننا سنقوم بتعديل قيمة الزمن وجعله يساوي ٢ وسنقوم بتشغيل النموذج.



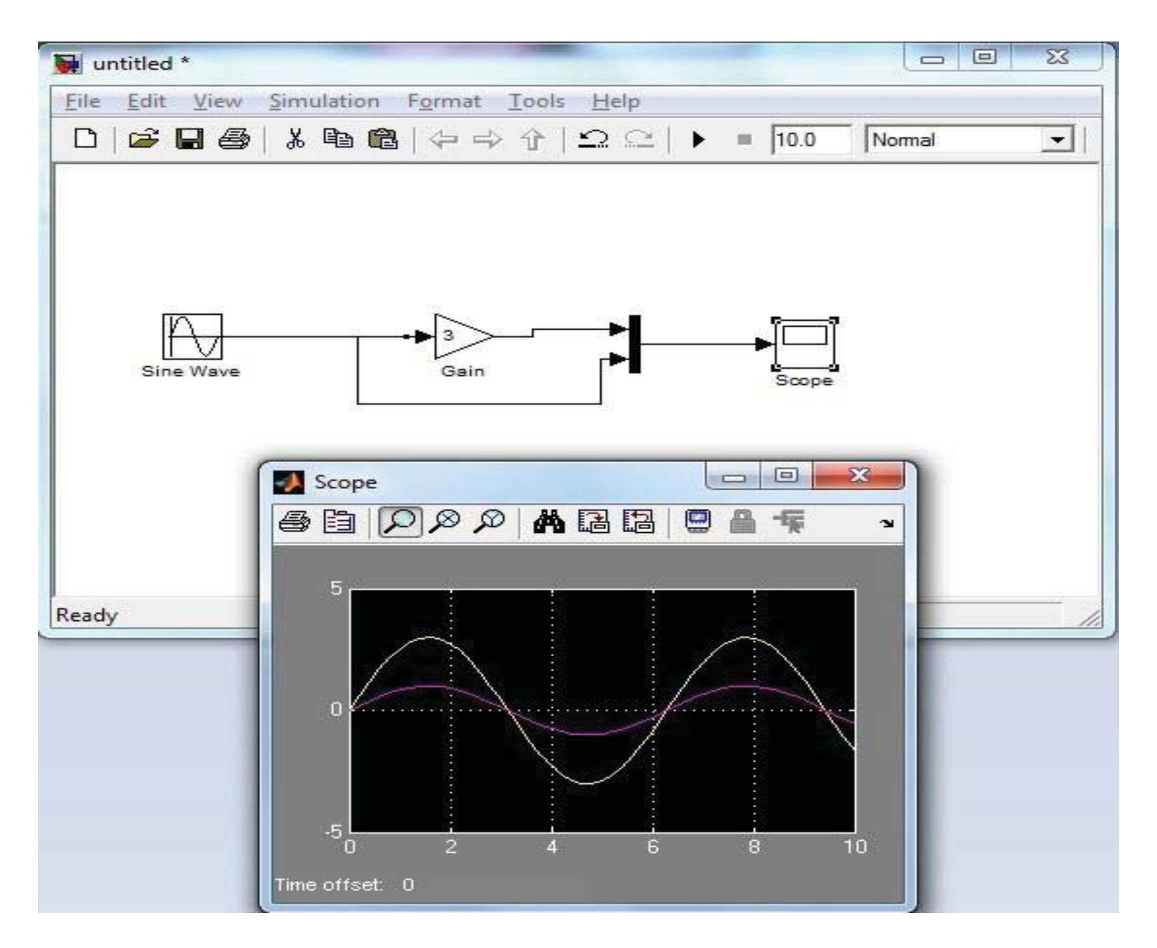
مثال (٢) :

رسم تغيرات موجة جيبية (sin) بعد ضربها بثابت k ومقارنة شكل الإشارة قبل وبعد ضربها بالثابت.

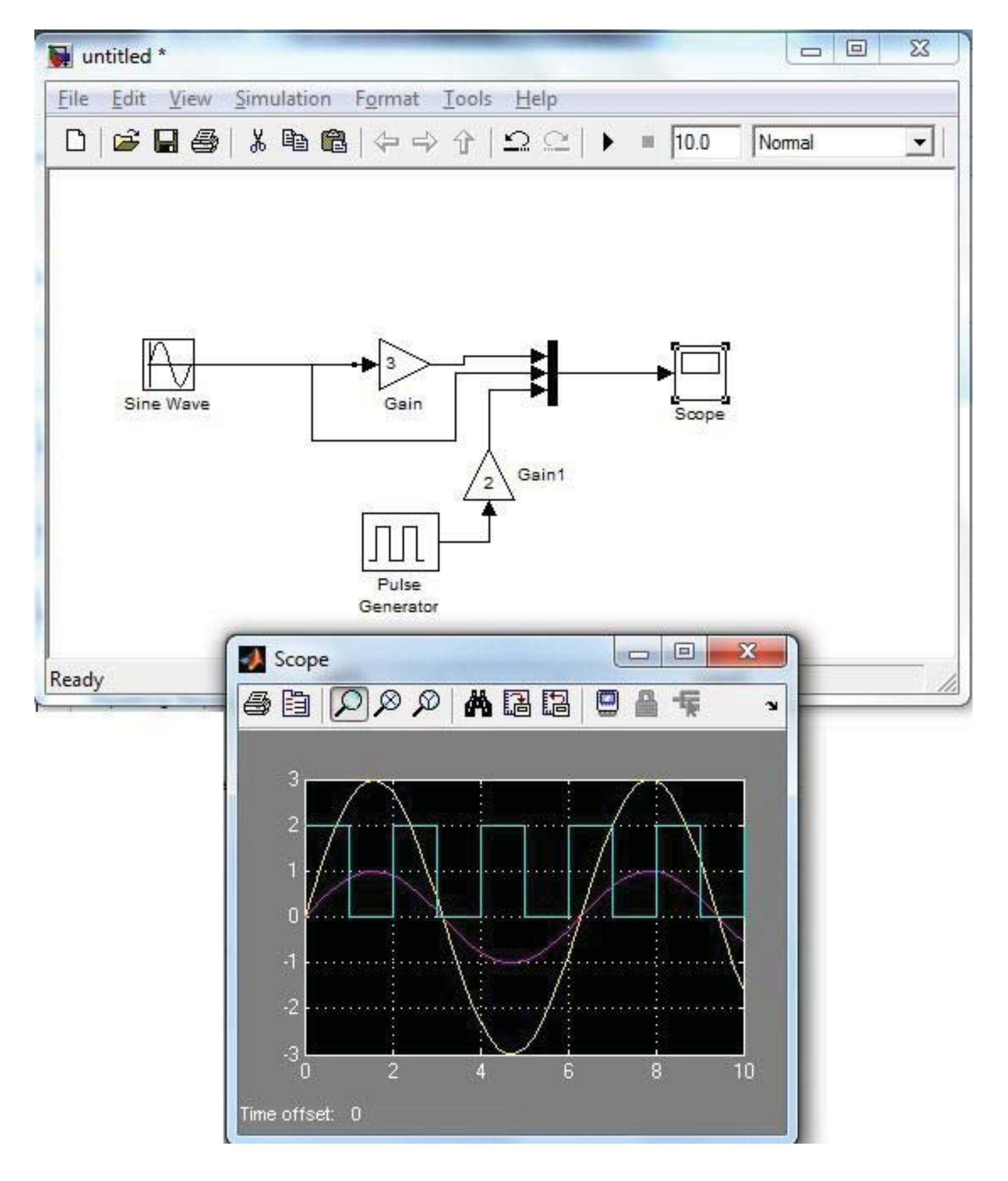
- نحتاج في هذا المثال إلى العناصر التالية :
  ١- منبع إشارة sinwave.
  ٢- عنصر تكبير Gain (يتم تعديل قيمة الثابت فيه من خصائصه)
  ٣- Scope لرسم الإشارة.

٤- Mux (عنصر دمج للإشارات ، نتحكم بعدد المداخل من خصائصه)

بعد إنشاء النموذج والتعديل في قيمة الثابت K وتشغيل النموذج نلاحظ من الشكل التالي المخطط و الرسم البياني الذي يحوي مقارنة بين الإشارة الأصلية والإشارة المضروبة بالثابت k=3.



لو أضفنا للنموذج السابق مولد نبضات وقمنا بتشغيل النموذج يكون الشكل كالتالي :



تعرفنا في هذا الفصل على بعض المكاتب الموجودة في الماتلاب ولاحظنا من خلال بعض الأمثلة البسيطة كيف يمكن بناء النموذج وكيف يمكن الحصول على نتائجه باستخدام البيئة Simulink في الماتلاب.

نمذجة و محاكاة نماذج بسيطة لأنظمة هندسية

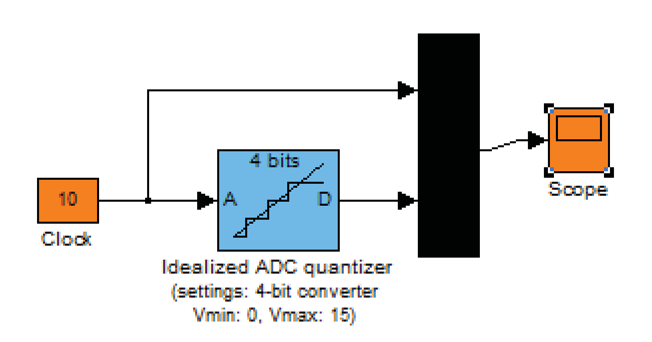
تمثيل نظم التحكم

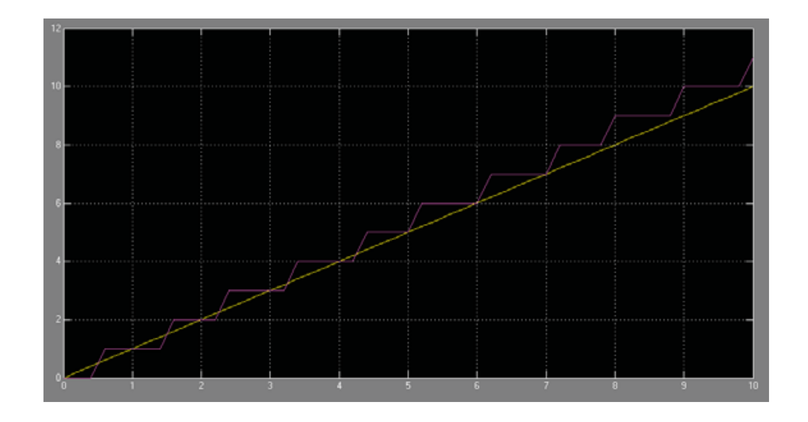
١,١٣ مقدمة

١,١,١٣ المبدل التمثيلي-الرقمي Analog-to-Digital Converter

أحد أهم البلوكات الجاهزة التي تمت إضافتها في الإصدارات الجديدة لمكتبة (simulink) هي المبدل التمثيلي- الرقمي المثالي.

ولابد من الإشارة للانتباه لإعدادات المبدل (البلوك) في الموديل حيث أن بيانات الخرج لكل من المبدل والساعة يجب أن تحدد بالنوع (double).



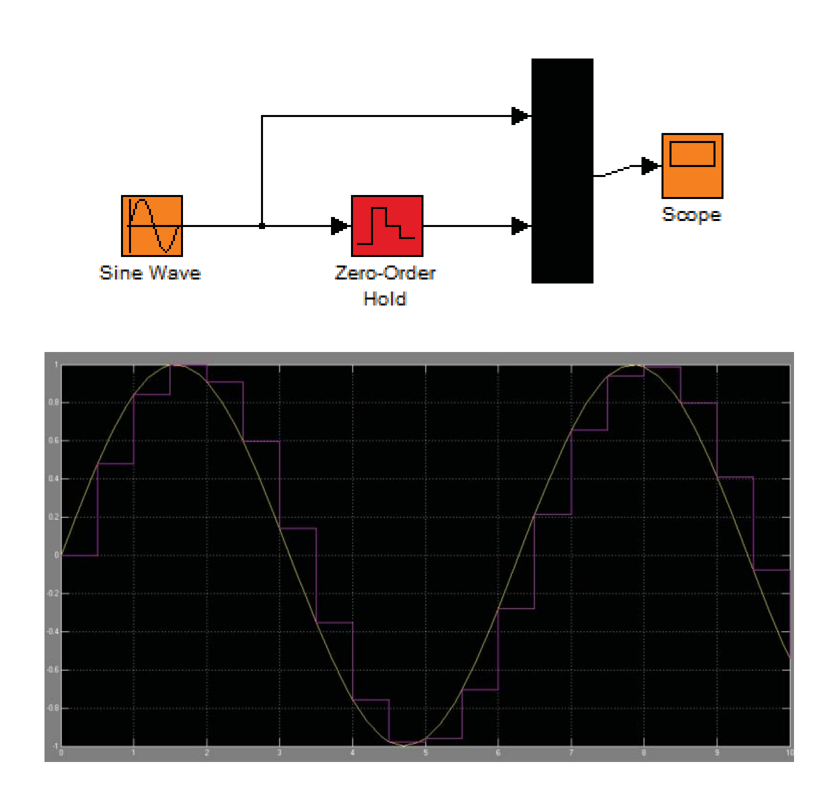


٢,١,١٣ ماسك النظام الصفري (Zero-order Hold) وماسك النظام الأول (First-order Hold)

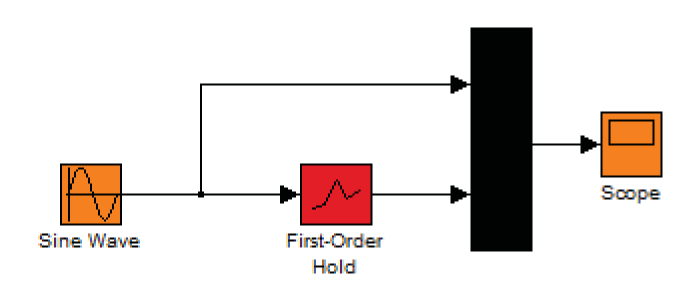
لنفرض أن لدينا إشارة مستمرة مع الزمن بعرض مجال محدود ويحدده عرض الحزمة (B) ، وليكن تحويل فورييه له |X(w)| مساوي للصفر من أجل أي قيمة لـ B < w.

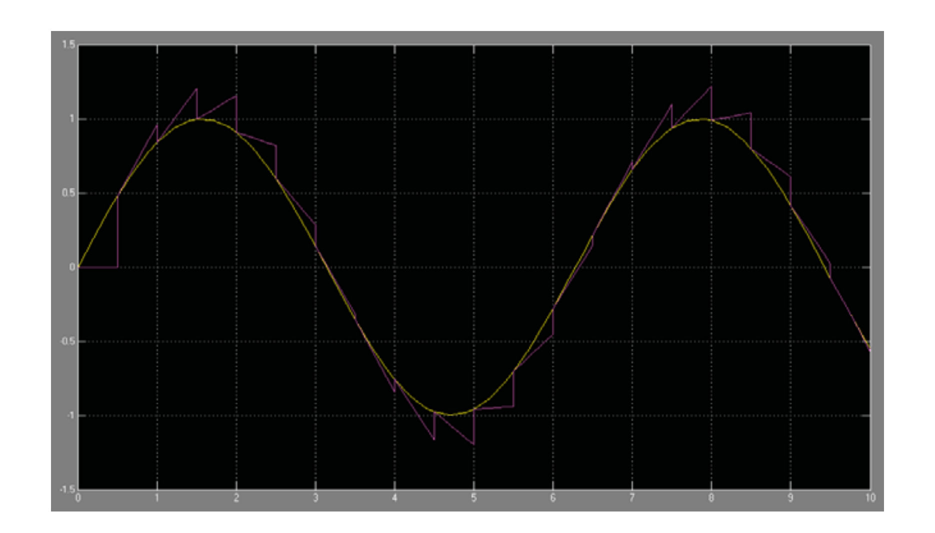
تنص نظرية أخذ العينات على أنه إذا كان تردد إشارة العينة w_s مساوي أو أكبر من 2B ، فإن الإشارة x(t) يمكن أن تعاد تماماً من الإشارة المعتانة $x_s(t)$ وذلك بتطبيق $x_s(t)$ على مرشح تمرير منخفض بعرض حزمة B. وأيضاً يمكن استعادة الإشارة الأصلية باستخدام دارة ماسك بحيث تمسك قيمة الإشارة المعتانة لزمن nT حتى تصل القيمة التالية في الزمن nT + T. إن سلوك ماسك الترتيب الصفري مشابه تماماً لسلوك مرشح

التمرير المنخفض ولذلك نستخدم هذه الدارة من أجل استعادة إشارة الزمن المستمرة من الإشارة المعتانة.



حيث أن ماسك الترتيب الصفري يولد إشارات دخل مستمرة u(t) بإمساك كل عينة لها قيمة ثابتة u[k] خلال زمن عينة واحدة ، أما دارة ماسك الترتيب الأول فتستعمل الاستيفاء الخطي بين العينات.

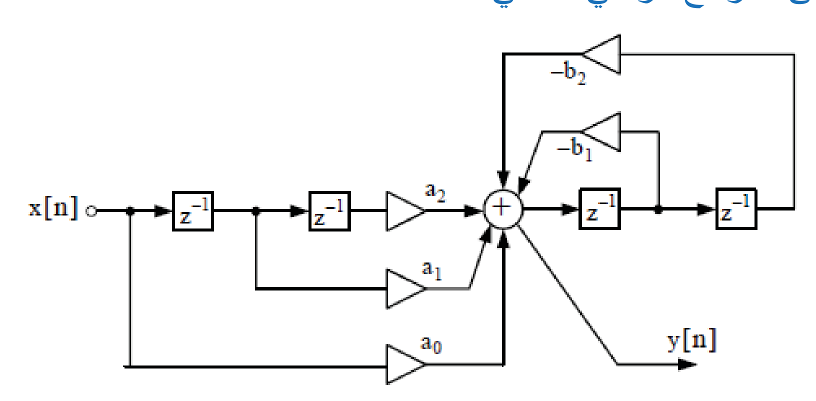




٣,١,١٣ التشكيلات المختلفة للمرشحات الرقمية (Digital Filter)

تابع النقل H(z) للمرشح الرقمي المثالي يعطى بعدة أشكال ، وأغلب الأشكال الشائعة هي التشكلية المباشرة الأولى والتشكيلة المباشرة الثانية والتوصيل التسلسلي والتوصيل التفرعي.

- التشكيلة الأولى للمرشح الرقمي المثالي :



$$a_0X(z) + a_1z^{-1}X(z) + a_2z^{-2}X(z) + (-b_1)z^{-1}Y(z) + (-b_2)z^{-1}Y(z) = Y(z)$$

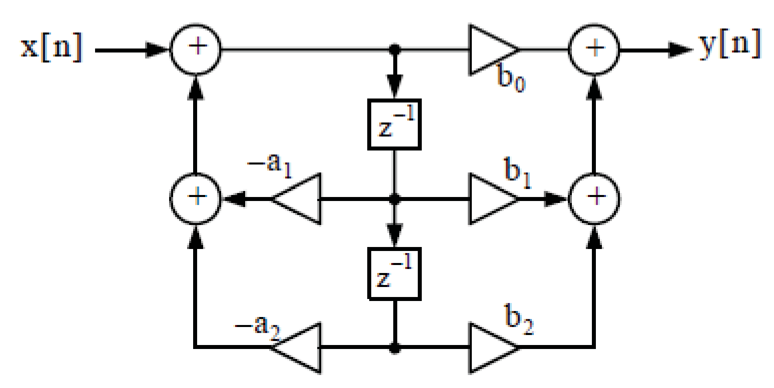
$$X(z)(a_0 + a_1z^{-1} + a_2z^{-2}) = Y(z)(1 + b_1z^{-1} + b_2z^{-2})$$

وبالنتيجة نحصل على تابع النقل للتشكلية الأولى للمرشح الرقمي المثالي وذلك بالعلاقة التالية

$$H(z) = \frac{Y(z)}{X(z)}$$

إن سيئة هذه التشكيلة من المرشحات الرقمية أنها تحتاج لسجلات 2k حيث k تمثل ترتيب الفلتر(المرشح).

- التشكيلة الثانية للمرشح الرقمي المثالي :



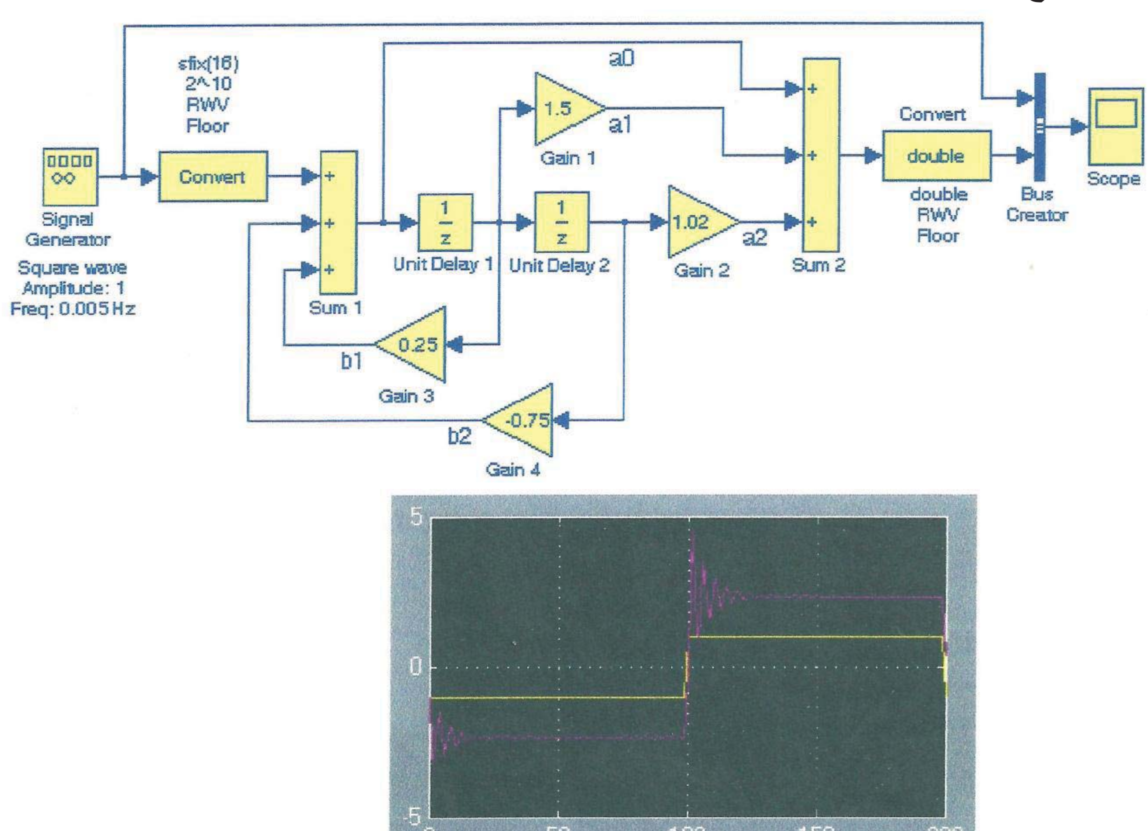
تتميز هذه التشكلية عن السابقة بأنها تحتاج فقط لسجل k يدل عليها بالعنصر z^{-1} ، حيث أن العنصر (z^{-1}) في التشكيلة الثانية يتوضع بين راسب و قطب.

مثال (١)

بين إشارتي الدخل والخرج للمرشح الرقمي الذي له تابع النقل :

$$H(z) = \frac{1+1.5z^{-1}+1.02z^{-2}}{1-0.25z^{-1}+0.75z^{-2}}$$

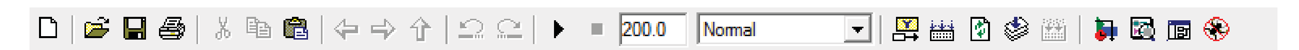
وذلك بوضع مخطط بالاستعانة بالمكتبة (Simulink) يمثل تابع النقل السابق.
لا تنسى أن تضع زمن التنفيذ ٢٠٠.

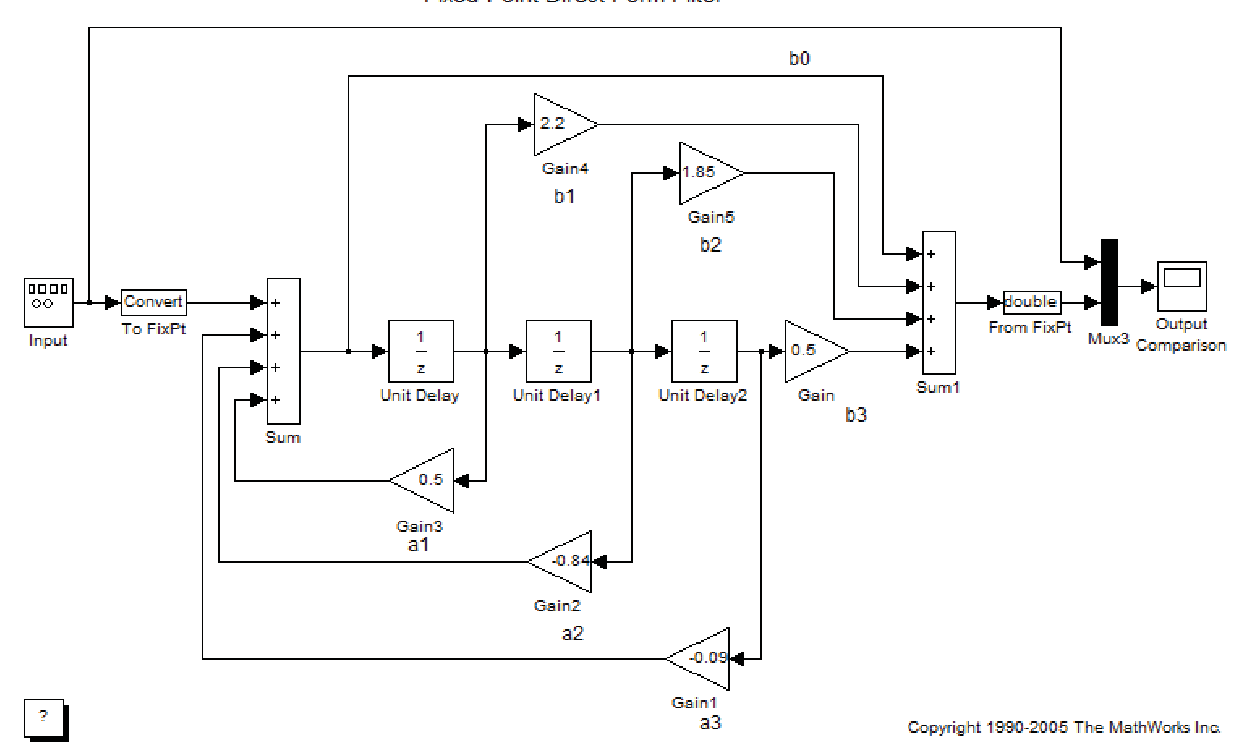


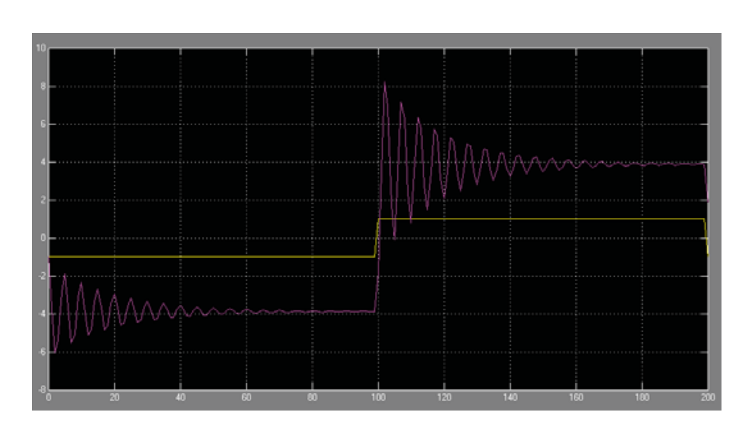
مثال (٢)

بين إشارتي الدخل والخرج للمرشح الرقمي الذي له تابع النقل

$$H(z) = \frac{1+2.2z^{-1}+1.85z^{-2}+0.5z^{-3}}{1-0.5z^{-1}+0.84z^{-2}+0.09z^{-3}}$$







اكتب بسطر الأوامر في الماتلاب : fxpdemo_direct_form2 لتحصل على المخطط الأخير.

• الوصل التسلسلي للمرشحات الرقمية :

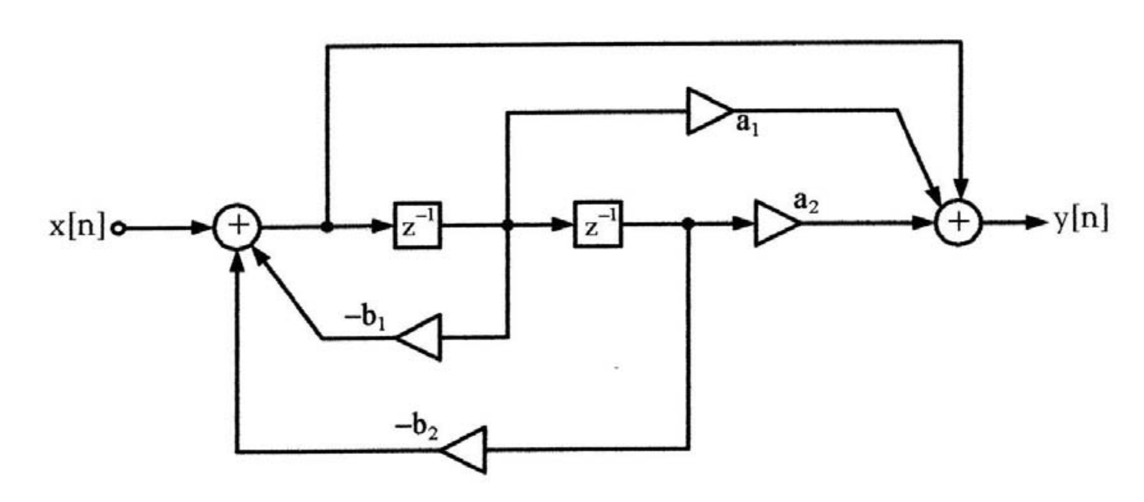
من أجل التوصيل التسلسلي للمرشحات الرقمية يعطى تابع النقل لهذه التشكلية بالعلاقة التالية :

$$H(z) = H_1(z).H_2(z) .... H_R(z)$$

X(z) → $H_1(z)$ → $H_2(z)$ → - - - → $H_R(z)$ → Y(z)

والشكل التالي يبين المخطط الصندوقي لنظام مرشح رقمي من الدرجة الثانية تابع النقل له هو :

$$H(z) = \frac{1+a_1z^{-1}+a_2z^{-2}}{1+b_1z^{-1}+b_2z^{-2}}$$



مثال (١) :

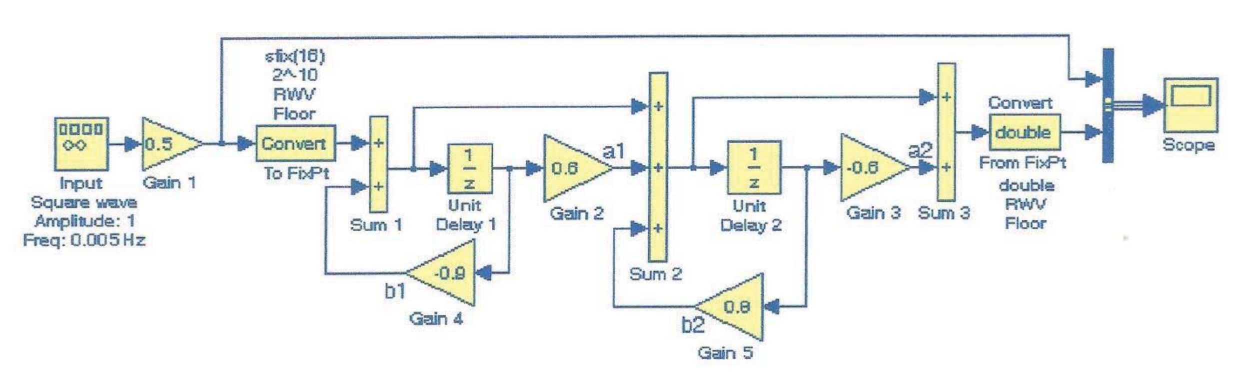
بين إشارتي الدخل والخرج لمرشح رقمي موصل بشكل تسلسلي له تابع النقل التالي

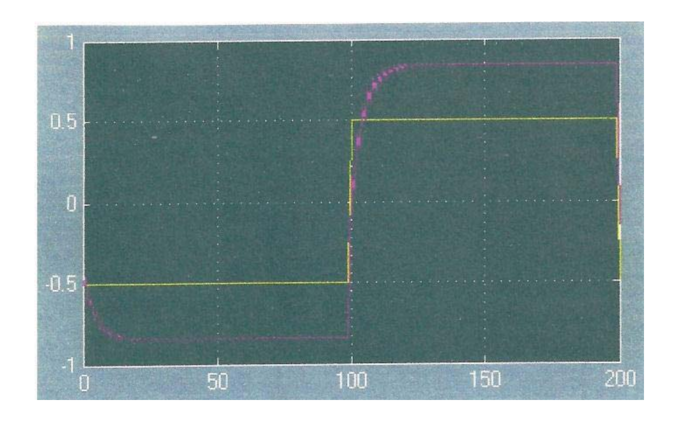
$$H(z) = \frac{0.5(1-0.36z^{-2})}{1+0.1z^{-1}-0.72z^{-2}}$$

الحل :

بعد إعادة تنسيق تابع النقل وتبسيطه نحصل على الشكل التالي :

$$H(z) = \frac{0.5(1+0.6z^{-1})(1-0.6z^{-1})}{(1+0.9z^{-1})(1-0.8z^{-1})}$$





ملاحظة :

بكتابة الأمر التالي في سطر الأوامر في الماتلاب :

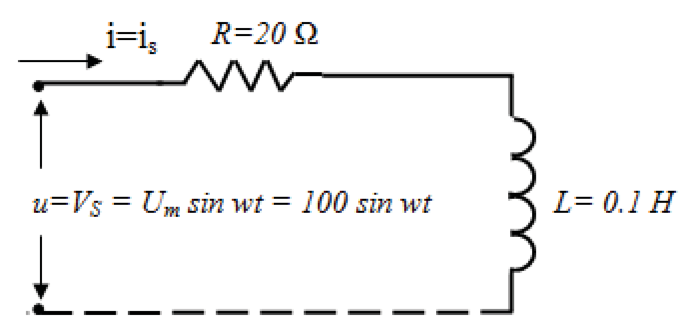
fxpdemo_series_cascade_form نحصل على المخطط الصندوقي للمرشح الرقمي التسلسلي الذي له تابع النقل

$$H(z) = \frac{(1+0.5z^{-1})(1+1.7z^{-1}+z^{-2})}{(1+0.1z^{-1})(1-0.6z^{-1}+0.9z^{-2})}$$

تمثيل ومحاكاة الدارات الكهربائية

٢,١٣ النمذجة و المحاكاة لدارة كهربائية تسلسلية RLC

سنبدأ الشرح بمثال بسيط ومن ثم سنعطي أمثلة أكثر تعقيداً و هكذا :



المطلوب وضع المخطط الصندوقي لمحاكاة دارة كهربائية تسلسلية بسيطة مكونة من مقاومة وملف ومنبع تغذية متناوب ورسم القيم العظمى للتوتر وللتيار والاستطاعة حيث :

R=20 Ω   L=0.1 H

خطوات الحل :

١- استنتاج المعادلة التفاضلية الواصفة للدارة (النموذج الرياضي الواصف للنظام).
٢- رسم المخطط الصندوقي اللازم لإجراء المحاكاة.
٣- إدخال معطيات النظام (تنفذ بعدة طرق).

الحل :

بما أن الدارة تسلسلية فإن معادلة الجهد الواصفة للنظام هي : $u = u_1 + u_2 = R.i + L\frac{di}{dt}$

حيث :

- u_1 التوتر الهابط على المقاومة R.
- u_2 التوتر الهابط على المحارضة L.
- U التوتر المتناوب المطبق على الدارة الكهربائية

نعيد كتابة المعادلة التفاضلية لتصبح بالشكل : $\frac{di}{dt} = \frac{1}{L}u - \frac{R}{L}i$

ونحصل على قيمة شدة التيار المار في الدارة بإجراء التكامل للمعادلة التفاضلية و فرض أن القيمة الابتدائية للتيار $i_o=0$ ، نحصل على النموذج الرياضي النهائي اللازم لإنشاء مخطط المحاكاة للدارة الكهربائية المفروضة (تسلسلية بسيطة) :

$$i(t) = \int \left(\frac{1}{L}u - \frac{R}{L}i\right) dt + i_o$$

ويمكن إجراء تعديل بسيط على المعادلة الأخيرة لتصبح :

$$i(t) = \frac{1}{L}\int (u - Ri)dt + i_o$$

والجدول التالي يبين المعادلات الواصفة للنظام والتي سنبني المخطط تبعاً لها

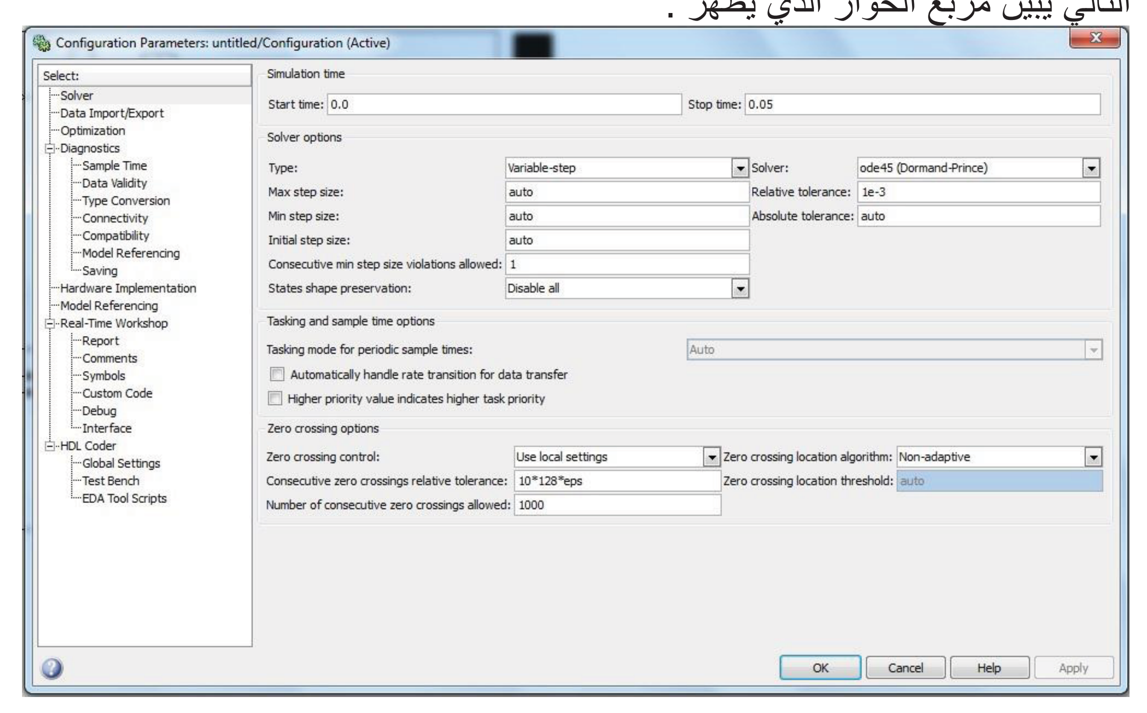
$\mathbf{i(t) = \frac{1}{L}\int (u - Ri)dt + i_o}$
$\mathbf{u_1 = R.i}$
$\mathbf{u_2 = L\frac{di}{dt}}$
$\mathbf{u = u_1 + u_2 = R.i + L\frac{di}{dt}}$

في مثالنا هذا نحتاج إلى العناصر التالية :

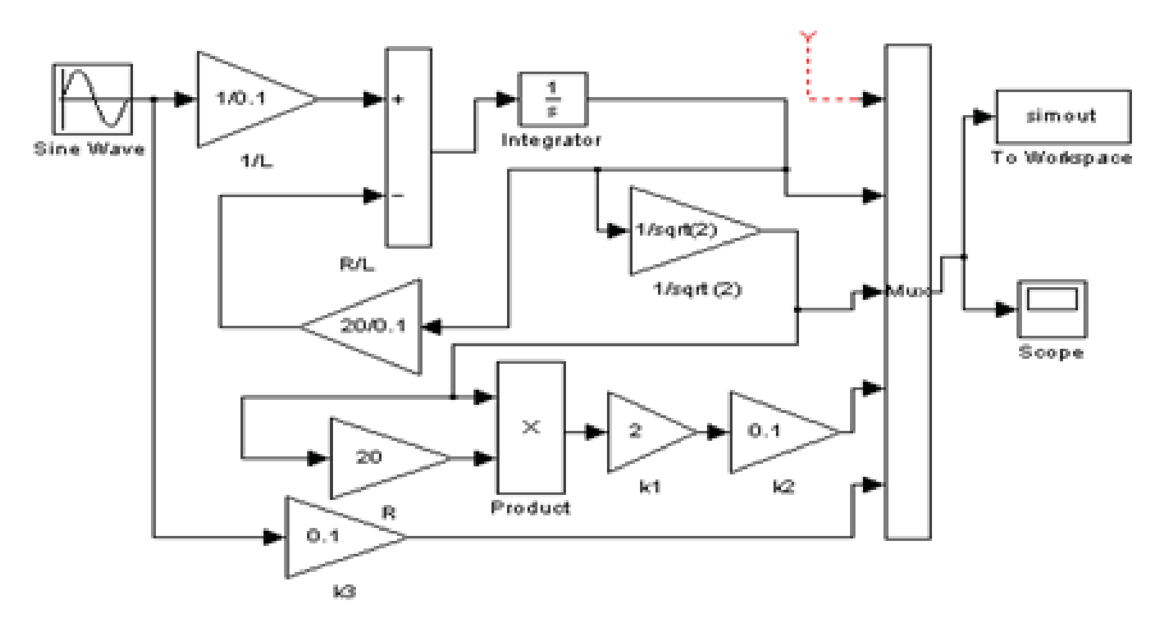
- عنصر تكامل Integrator ويؤخذ من المكتبة Continuous.
- مكبر Gain ويؤخذ من المكتبة Math.
- عنصر جداء Product ويؤخذ من المكتبة Math.
- عنصر الجمع sum ويؤخذ من المكتبة Math.
- عنصر مزج Mux ويؤخذ من المكتبة signals & system
- منبع متناوب جيبي sinwave وتؤخذ من المكتبة Sources.
- عداد الزمن أو إشارة الزمن ويؤخذ من المكتبة Sources.
- وسيلة إظهار النتائج Scope ويؤخذ من المكتبة sinks.
- عنصر إخراج النتائج بشكل رقمي على صفحة عمل الماتلاب بشكل مصفوفة (to workspace) ويؤخذ من المكتبة sinks.
- إشارة ثابتة أو قيمة ثابتة Constant وتؤخذ من المكتبة sources.

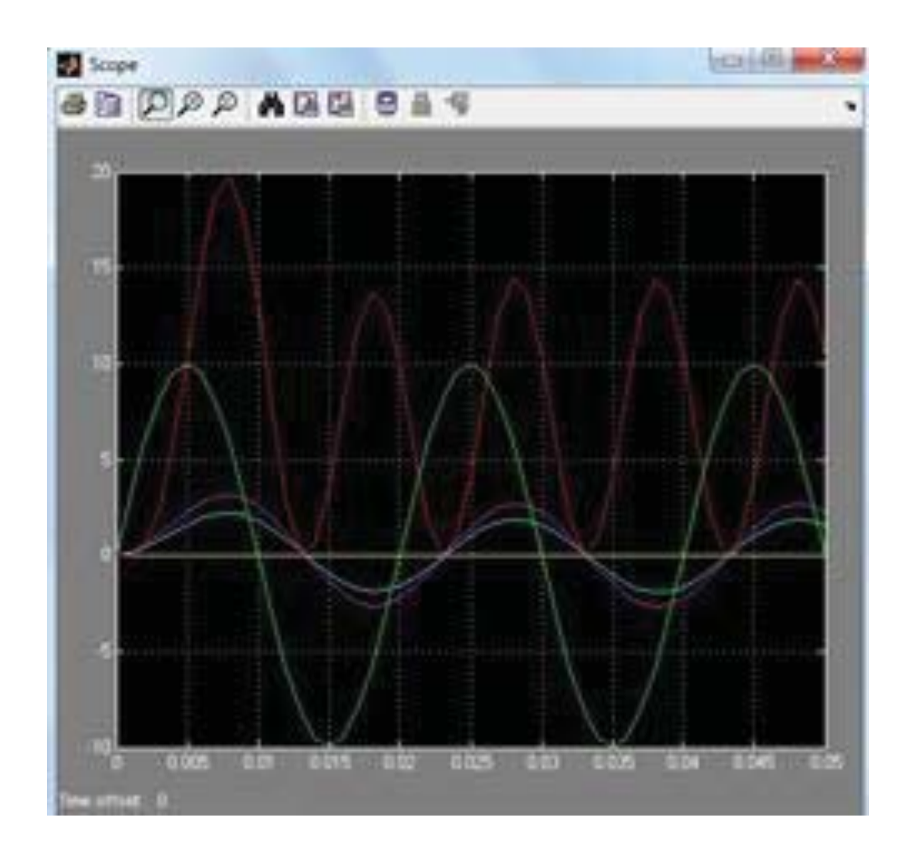
الآن بعد أن تم وضع جميع العناصر اللازمة على الصفحة (model) يتم ربط العناصر مع بعضها ، بما يتناسب مع المعادلة التفاضلية الواصفة للدارة ، بحيث نحصل في النهاية

على النموذج الرياضي النهائي جاهزاً لإجراء النمذجة.
لاختيار زمن المحاكاة وليكن (0.05sec) مثلاً ، يتم ذلك من خلال الأمر simulation في شريط الأدوات ومنه اختيار الأمر Configuration Parameters ، حيث صندوق الحوار الناتج يحوي مربع إمكانية تغيير زمن المحاكاة واختيار طريقة التكامل ، والشكل التالي يبين مربع الحوار الذي يظهر .



وبعد تحديد زمن التنفيذ ٠,٠٥ ثانية ومن ثم تحديد نوع التكامل والضغط على موافق ومن ثم تشغيل النموذج نجد أن نتائج المحاكاة للدارة الكهربائية هي كما يبين الشكل التالي :

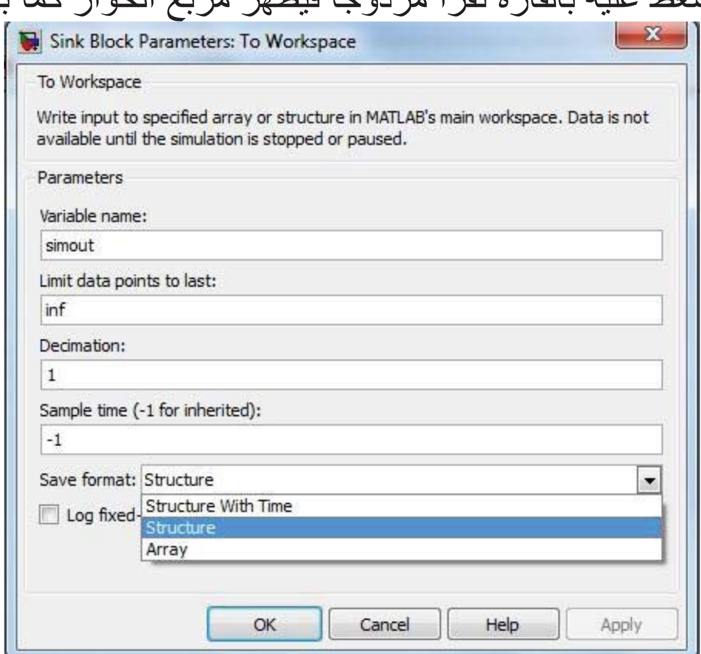




١,٢,١٣ طرائق إظهار نتائج المحاكاة :

١- **الإظهار مباشرة على Scope :** هذه الطريقة تعطي تصوراً مبدئياً عن تغيرات الإشارة ولا تعطي التفصيل الخاصة .

٢- **إرسال النتائج إلى نافذة الأوامر to workspace :** يمكن التحكم بخصائص هذا العنصر بالضغط عليه بالفأرة نقراً مزدوجاً فيظهر مربع الحوار كما بالشكل :



ما يهم من هذه الخصائص هو طريقة الحفظ إلى نافذة الأوامر فهي يمكن أن تكون إما مصفوفة أو بنية (structure) أو بنية متغيرة مع الزمن. لاحظ لو أننا اخترنا حفظ على شكل مصفوفة ومن ثم شغلنا النموذج وبعد ذلك توجهنا نحو نافذة الأوامر وكتبنا simout (variable name) لوجدنا النتائج كما يبين الشكل التالي :

Shortcuts How to Add What's New

Current Directory

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
>> simout

simout =

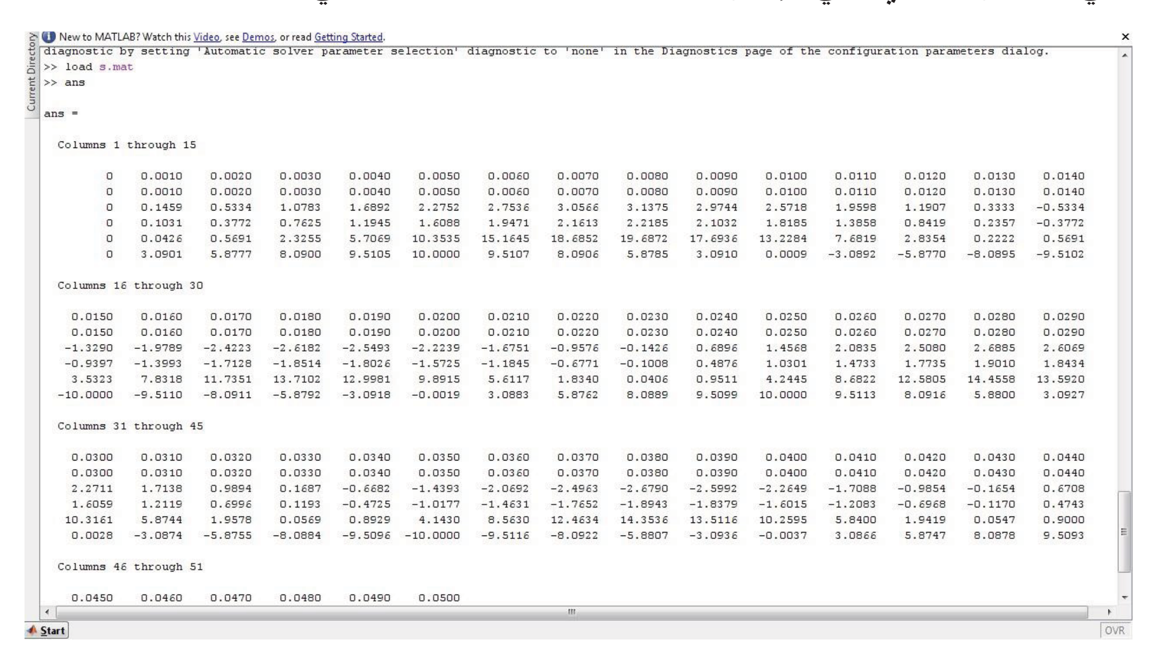
         0         0         0         0         0
    0.0010    0.1459    0.1031    0.0426    3.0901
    0.0020    0.5334    0.3772    0.5691    5.8777
    0.0030    1.0783    0.7625    2.3255    8.0900
    0.0040    1.6892    1.1945    5.7069    9.5105
    0.0050    2.2752    1.6088   10.3535   10.0000
    0.0060    2.7536    1.9471   15.1645    9.5107
    0.0070    3.0566    2.1613   18.6852    8.0906
    0.0080    3.1375    2.2185   19.6872    5.8785
    0.0090    2.9744    2.1032   17.6936    3.0910
    0.0100    2.5718    1.8185   13.2284    0.0009
    0.0110    1.9598    1.3858    7.6819   -3.0892
    0.0120    1.1907    0.8419    2.8354   -5.8770
    0.0130    0.3333    0.2357    0.2222   -8.0895
    0.0140   -0.5334   -0.3772    0.5691   -9.5102
    0.0150   -1.3290   -0.9397    3.5323  -10.0000
    0.0160   -1.9789   -1.3993    7.8318   -9.5110
    0.0170   -2.4223   -1.7128   11.7351   -8.0911
    0.0180   -2.6182   -1.8514   13.7102   -5.8792
    0.0190   -2.5493   -1.8026   12.9981   -3.0918
    0.0200   -2.2239   -1.5725    9.8915   -0.0019
    0.0210   -1.6751   -1.1845    5.6117    3.0883
    0.0220   -0.9576   -0.6771    1.8340    5.8762
    0.0230   -0.1426   -0.1008    0.0406    8.0889
    0.0240    0.6896    0.4876    0.9511    9.5099
    0.0250    1.4568    1.0301    4.2445   10.0000
    0.0260    2.0835    1.4733    8.6822    9.5113
    0.0270    2.5080    1.7735   12.5805    8.0916
```

نلاحظ أن هذا البلوك قام بحفظ النتائج الظاهرة لدينا كمصفوفة يمكننا التعامل معها كيفما نشاء ، حيث أن كل عمود من أعمدة المصفوفة يمثل متغير محدد كالجهد أو التيار أو الاستطاعة ....إلخ

٣- **إرسال النتائج إلى ملف ذي امتداد mat :** نسميه مثلاً بالاسم (s.mat)
ولاستدعاء النتائج المخزنة في هذا الملف بعد انتهاء المحاكاة نعود إلى نافذة الماتلاب الرئيسية ونكتب :

```
>> Load s.mat
```

فيعرض لنا الحاسب النتائج على شكل مصفوفة سطرها الأول يحتوي على الزمن أما باقي الأسطر فتحوي باقي الإشارات ، وذلك كما يبين الشكل التالي :



```
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
diagnostic by setting 'Automatic solver parameter selection' diagnostic to 'none' in the Diagnostics page of the configuration parameters dialog.
>> load s.mat
>> ans

ans =

  Columns 1 through 15

         0    0.0010    0.0020    0.0030    0.0040    0.0050    0.0060    0.0070    0.0080    0.0090    0.0100    0.0110    0.0120    0.0130    0.0140
         0    0.0010    0.0020    0.0030    0.0040    0.0050    0.0060    0.0070    0.0080    0.0090    0.0100    0.0110    0.0120    0.0130    0.0140
         0    0.1459    0.5334    1.0783    1.6892    2.2752    2.7536    3.0566    3.1375    2.9744    2.5718    1.9598    1.1907    0.3333   -0.5334
         0    0.1031    0.3772    0.7625    1.1945    1.6088    1.9471    2.1613    2.2185    2.1032    1.8185    1.3858    0.8419    0.2357   -0.3772
         0    0.0426    0.5691    2.3255    5.7069   10.3535   15.1645   18.6852   19.6872   17.6936   13.2284    7.6819    2.8354    0.2222    0.5691
         0    3.0901    5.8777    8.0900    9.5105   10.0000    9.5107    8.0906    5.8785    3.0910    0.0009   -3.0892   -5.8770   -8.0895   -9.5102

  Columns 16 through 30

    0.0150    0.0160    0.0170    0.0180    0.0190    0.0200    0.0210    0.0220    0.0230    0.0240    0.0250    0.0260    0.0270    0.0280    0.0290
    0.0150    0.0160    0.0170    0.0180    0.0190    0.0200    0.0210    0.0220    0.0230    0.0240    0.0250    0.0260    0.0270    0.0280    0.0290
   -1.3290   -1.9789   -2.4223   -2.6182   -2.5493   -2.2239   -1.6751   -0.9576   -0.1426    0.6896    1.4568    2.0835    2.5080    2.6885    2.6069
   -0.9397   -1.3993   -1.7128   -1.8514   -1.8026   -1.5725   -1.1845   -0.6771   -0.1008    0.4876    1.0301    1.4733    1.7735    1.9010    1.8434
    3.5323    7.8318   11.7351   13.7102   12.9981    9.8915    5.6117    1.8340    0.0406    0.9511    4.2445    8.6822   12.5805   14.4558   13.5920
  -10.0000   -9.5110   -8.0911   -5.8792   -3.0918   -0.0019    3.0883    5.8762    8.0889    9.5099   10.0000    9.5113    8.0916    5.8800    3.0927

  Columns 31 through 45

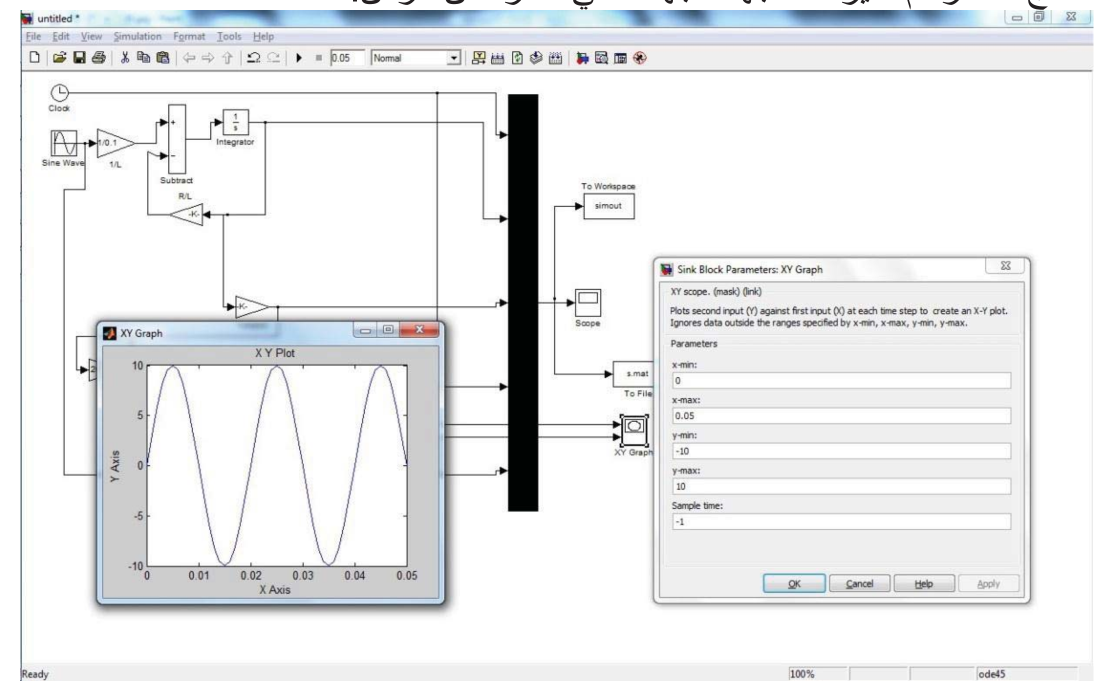
    0.0300    0.0310    0.0320    0.0330    0.0340    0.0350    0.0360    0.0370    0.0380    0.0390    0.0400    0.0410    0.0420    0.0430    0.0440
    0.0300    0.0310    0.0320    0.0330    0.0340    0.0350    0.0360    0.0370    0.0380    0.0390    0.0400    0.0410    0.0420    0.0430    0.0440
    2.2711    1.7138    0.9894    0.1687   -0.6682   -1.4393   -2.0692   -2.4963   -2.6790   -2.5992   -2.2649   -1.7088   -0.9854   -0.1654    0.6708
    1.6059    1.2119    0.6996    0.1193   -0.4725   -1.0177   -1.4631   -1.7652   -1.8943   -1.8379   -1.6015   -1.2083   -0.6968   -0.1170    0.4743
   10.3161    5.8744    1.9578    0.0569    0.8929    4.1430    8.5630   12.4634   14.3536   13.5116   10.2595    5.8400    1.9419    0.0547    0.9000
    0.0028   -3.0874   -5.8755   -8.0884   -9.5096  -10.0000   -9.5116   -8.0922   -5.8807   -3.0936   -0.0037    3.0866    5.8747    8.0878    9.5093

  Columns 46 through 51

    0.0450    0.0460    0.0470    0.0480    0.0490    0.0500
```

٤- إظهار النتائج عن طريق العنصر(XY Graph) :

تتم إضافة العنصر بحيث يوصل X منه إلى الزمن أي إلى مخرج الزمن t والمدخل الثاني Y يوصل الإشارة التي نريد رسمها ثم نفعل العنصر بالضغط عليه (Click) ، والشكل التالي يوضح التعديلات الواجب القيام بها من خصائص العنصر و الشكل الناتج عند رسم تغيرات الجهد للجهد الكلي للدارة من الزمن.



٥- إظهار النتائج باستخدام العنصر Display :

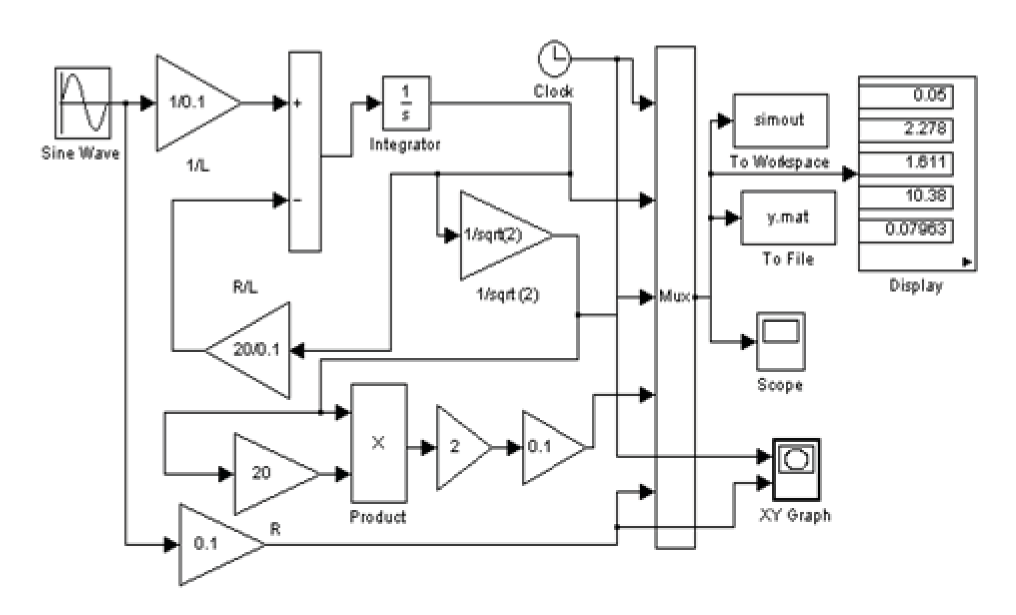
هناك إمكانية زود بها البرنامج من أجل إظهار القيم في أثناء إجراء عملية المحاكاة بشكل أرقام على شاشة العنصر ، حيث تتم مراقبة التغيرات في قيم الإشارة ويتوقف عرض النتائج في نهاية عملية المحاكاة عند آخر قيمة للزمن والقيم الموافقة لها من الإشارات.

٦- إظهار النتائج بطرائق أخرى :

١- استعمال الملفات ، يمكن أن يشكل ملف لقراءة المحددات للدارة ، وملف آخر لإظهار النتائج ، ويعطى كل ملف وحده اسماً منفصلاً.

٢- تشكيل ملف واحد لقراءة محددات الدارة وإخراج النتائج، تعد هذه الطريقة من أفضل النتائج المتبعة في عملية المحاكاة.

وبالنتيجة أصبح لدينا الشكل النهائي للنموذج كالتالي :

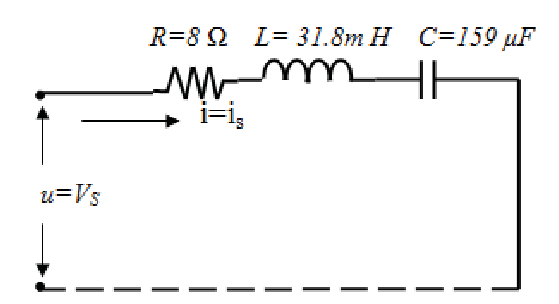


مثال غير محلول :

ليكن لدينا الدارة الكهربائية المكونة من العناصر R,L,C على التسلسل ، قيم معاملاتها كما هو مبين بالشكل والمطلوب :

آ : وضع المخطط الصندوقي اللازم لمحاكاة الدارة وحساب التيار المار فيها وإظهاره.

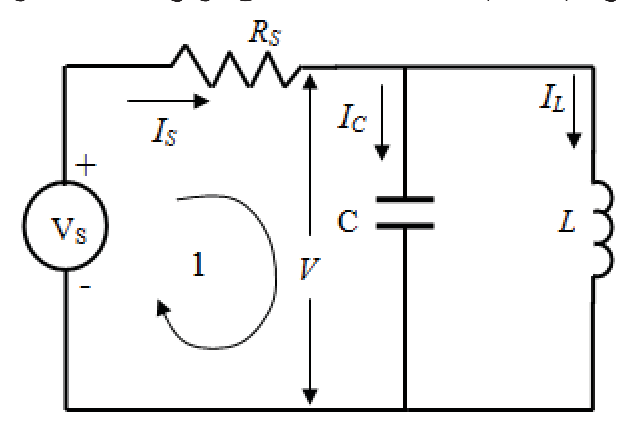
ب : كتابة ملف لإدخال المعطيات وإخراج النتائج.



٣,١٣ النمذجة و المحاكاة لدارة كهربائية تسلسلية - تفرعية RLC

مثال(1)

ليكن لدينا الدارة الكهربائية المبينة بالشكل ، علماً أن توتر الدخل متناوب



f = 50Hz - C = 1000µf - L = 0.1 H - R_s = 50 Ω - V_s= 100v

والمطلوب إجراء النمذجة والمحاكاة للدارة ووضع النموذج الرياضي للدارة المكافئة للحصول على التيار الكلي وتيار كل فرع ، وكتابة ملف إدخال المعطيات.

i_s التيار الكلي.

i_c التيار المار في الفرع الذي يحوي سعة فقط.

i_L التيار المار في الفرع الذي يحوي محارضة فقط.

وبالنتيجة فإن التيار الكلي المار في الدارة هو : $i_S = i_C + i_L$

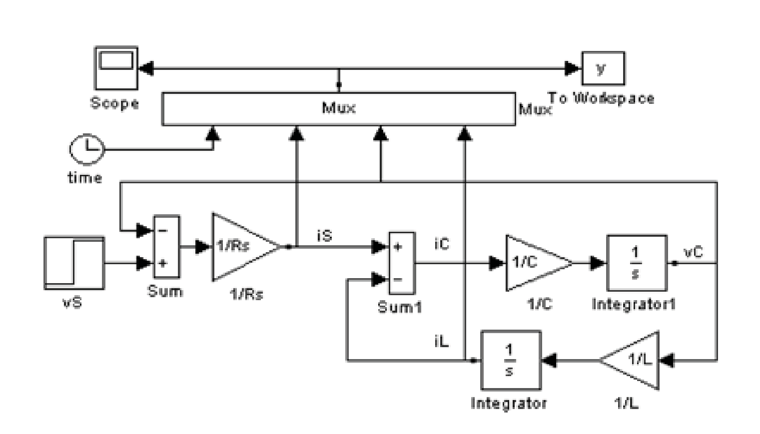
ومن تحليل الدارة المكافئة وحسب قانون كيرشوف (١) لدينا :

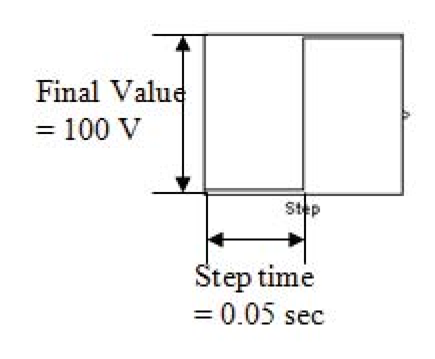
$$V_s = V_c + i_s.R_s$$

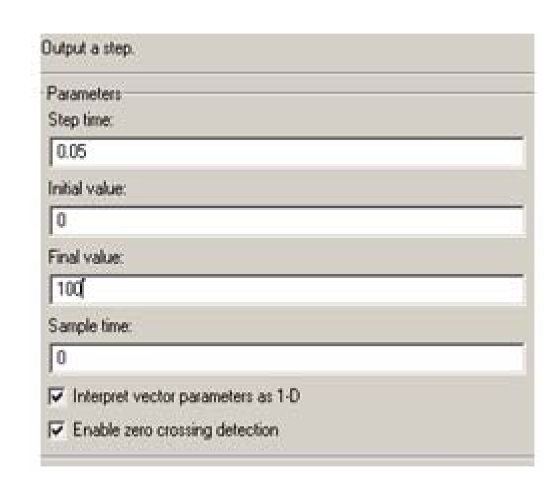
وحسب قانون كيرشوف أيضاً نجد : $i_s = i_L + i_c$

$$V_c = L.\frac{di}{dt} \quad \text{or} \quad V_c = \frac{1}{C}\int i_c\, dt$$

$$i_L = \frac{1}{L}\int V_c dt \quad \rightarrow \quad V_c = \frac{1}{C}\int\left(\frac{V_S - V_C}{R_S} - i_L\right)dt$$







نقوم بتعديل زمن التشغيل للموديل ليصبح ثانية واحدة.

- ملف إدخال المعطيات.

```
Rs=50;
L=0.1;
C=1000e-6;
VS_mag=100;
tdelay=0.05;
vCo=0;
iLo=0;
tstop=0.5;
disp('run simulation,type "return" when ready')
keyboard

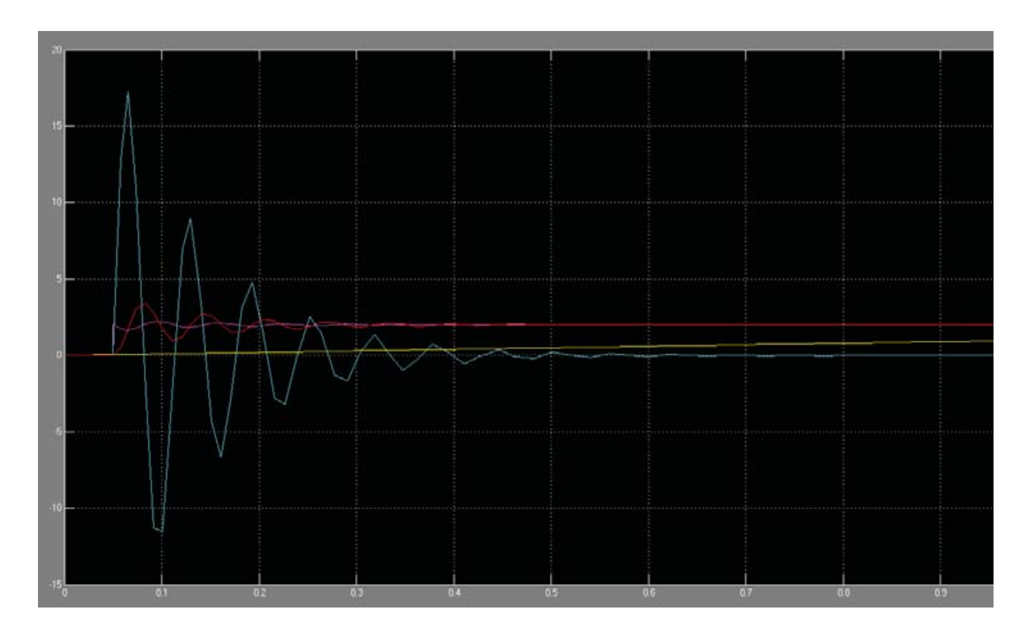
subplot(3,1,1)
plot(simout(:,1),simout(:,2),'k')
title('source current')
ylabel('iS in A')

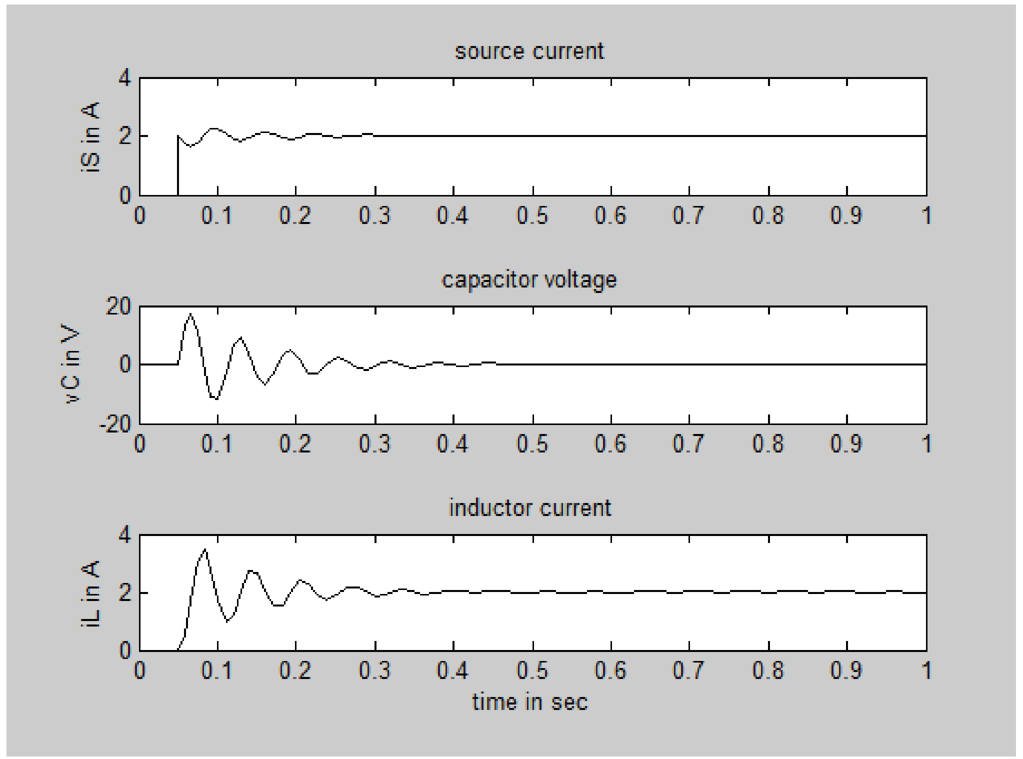
subplot(3,1,2)
plot(simout (:,1), simout (:,3),'k')
title('capacitor voltage')
ylabel('vC in V')

subplot(3,1,3)
plot(simout (:,1), simout (:,4),'k')
title('inductor current')
xlabel('time in sec')
ylabel('iL in A')
```

بعد كتابة الملف نقوم بتشغيله فنلاحظ أن قيم جميع المتحولات أصبح في workspace وبالتالي أصبح الآن بإمكاننا تشغيل الموديل.

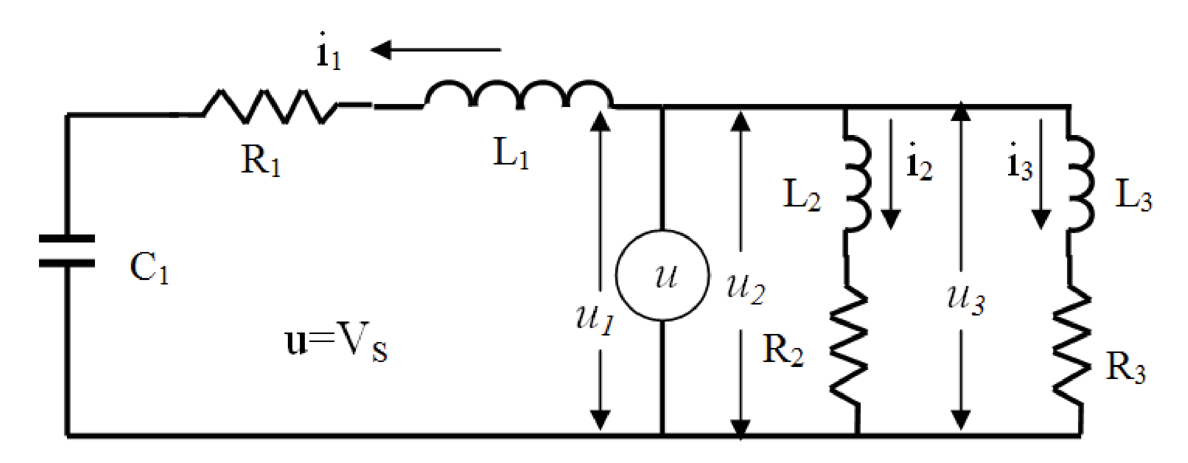
- بعد تشغيل الموديل (t=1) نلاحظ شكل أمواج التيار والتوتر التي تظهر على عنصر الإظهار scope هي كما يبين الشكل التالي .





مثال(2)

ليكن لدينا الدارة الكهربائية المبينة بالشكل ، علماً أن توتر الدخل متناوب :



$R_3=R_2=4\ \Omega$, $R_1=20\ \Omega$, $L_3=0.3\ H$,$L_2= 0.1\ H$, $L_1= 0.2\ H$
$F=50\ Hz$,$W=2\Pi f$, $U_{eff}=70/\sqrt{2}\ V$, $u=U\sin(wt)$,$C_1=1000\mu F$

- استنتاج النموذج الرياضي للدارة واللازم لحساب التيارات i_3 , i_2 , i_1.
- وضع مخطط المحاكاة اللازم لحساب قيم التيارات i_3 , i_2 , i_1 باستخدام برنامج Matlab .
- كتابة ملف لقراءة المعطيات وإظهار النتائج.

الحل:

- استنتاج المعادلة التفاضلية الواصفة للدارة (النموذج الرياضي) وهي:

$$u_1 = u_2 = u_3 = u$$

$$u_1 = R_1 i_1 + L_1 \frac{di_1}{dt} + \frac{1}{C_1}\int i_1 dt \Rightarrow \frac{di_1}{dt} = (\frac{u_1}{L_1} - \frac{R_1}{L_1} i_1 - \frac{1}{L_1 C_1}\int i_1 dt)$$

$$i_1 = \int (\frac{u_1}{L_1} - \frac{R_1}{L_1} i_1 - \frac{1}{L_1 C_1}\int i_1 dt)dt \quad or$$

$$i_1 = \frac{1}{L_1}\int (u_1 - R i_1 - \frac{1}{C_1}\int i_1 dt)dt$$

$$u_2 = R_2 i_2 + L_2 \frac{di_2}{dt} \Rightarrow \frac{di_2}{dt} = \frac{u_2}{L_2} - \frac{R_2}{L_2} i_2 \Rightarrow i_2 = \int (\frac{u_2}{L_2} - \frac{R_2}{L_2} i_2)\, dt \quad or$$

$$i_2 = \frac{1}{L_2}\int (u_2 - R_2 i_2)\, dt$$

$$u_3 = R_3 i_3 + L_3 \frac{di_3}{dt} \Rightarrow \frac{di_3}{dt} = \frac{u_3}{L_3} - \frac{R_3}{L_3} i_3 \Rightarrow i_3 = \int (\frac{u_3}{L_3} - \frac{R_3}{L_3} i_3)\, dt \quad or$$

$$i_3 = \frac{1}{L_3}\int (u_3 - R_3 i_3)\, dt$$

حيث :

التوتر المتناوب المطبق على الدارة الكهربائية $u=U_m\sin(wt)=70\sin(wt)$

التردد الزاوي بالراديان : $2\pi f$

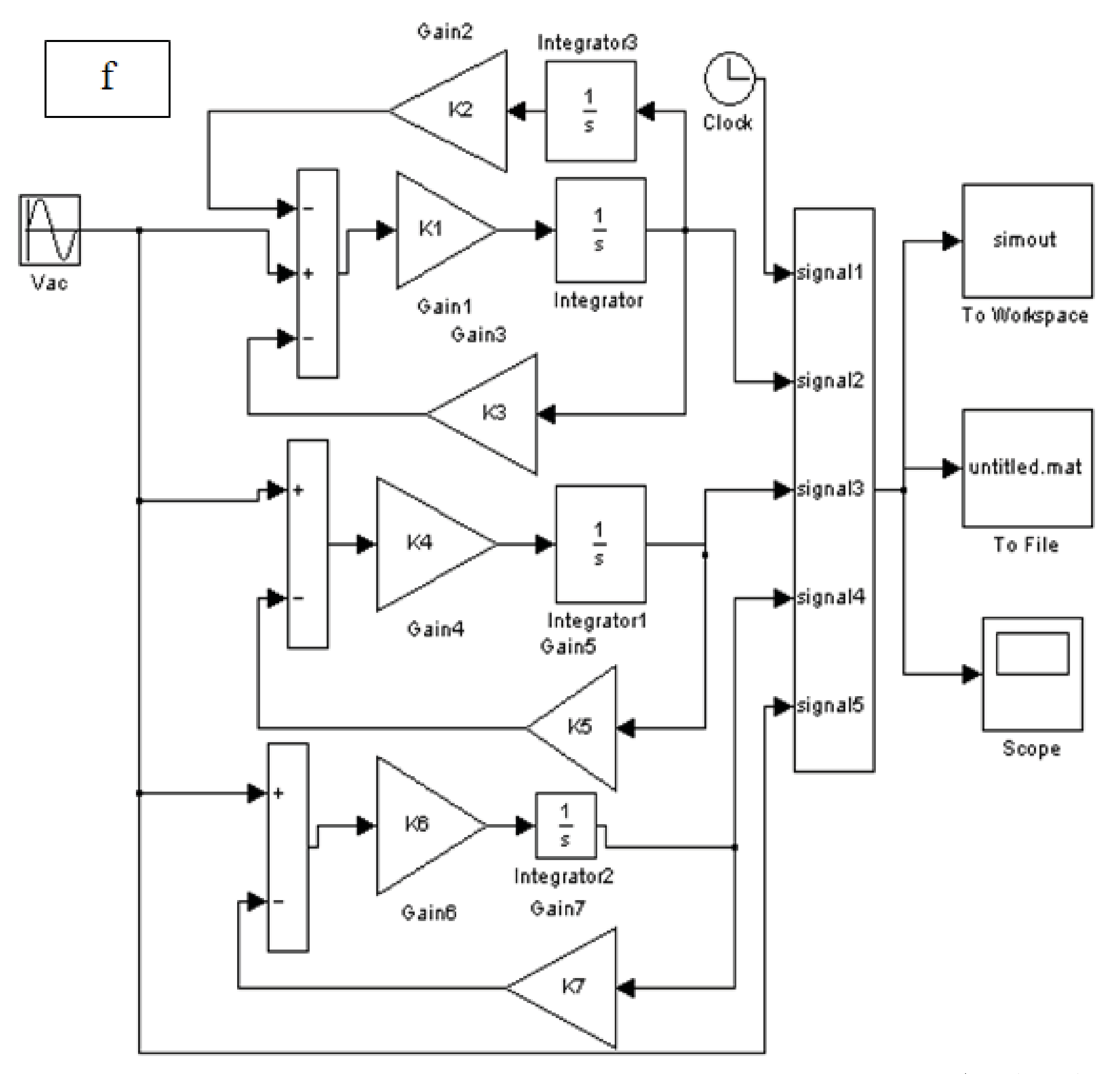
نحصل على النموذج الرياضي باعتبار القيمة الابتدائية للتيار تساوي الصفر $i_{(0)} = 0$ النهائي اللازم لإنشاء مخطط المحاكاة للدارة الكهربائية :

سوف نقوم بوضع المخطط الصندوقي اللازم للنمذجة اعتماداً على المعادلات

$$i_1 = \int\left(\frac{u_1}{L_1} - \frac{R_1}{L_1} i_1 - \frac{1}{L_1 C_1}\int i_1 dt\right)dt$$

$$i_2 = \int\left(\frac{u_2}{L_2} - \frac{R_2}{L_2} i_2\right)dt$$

$$i_3 = \int\left(\frac{u_3}{L_3} - \frac{R_3}{L_3} i_3\right)dt$$

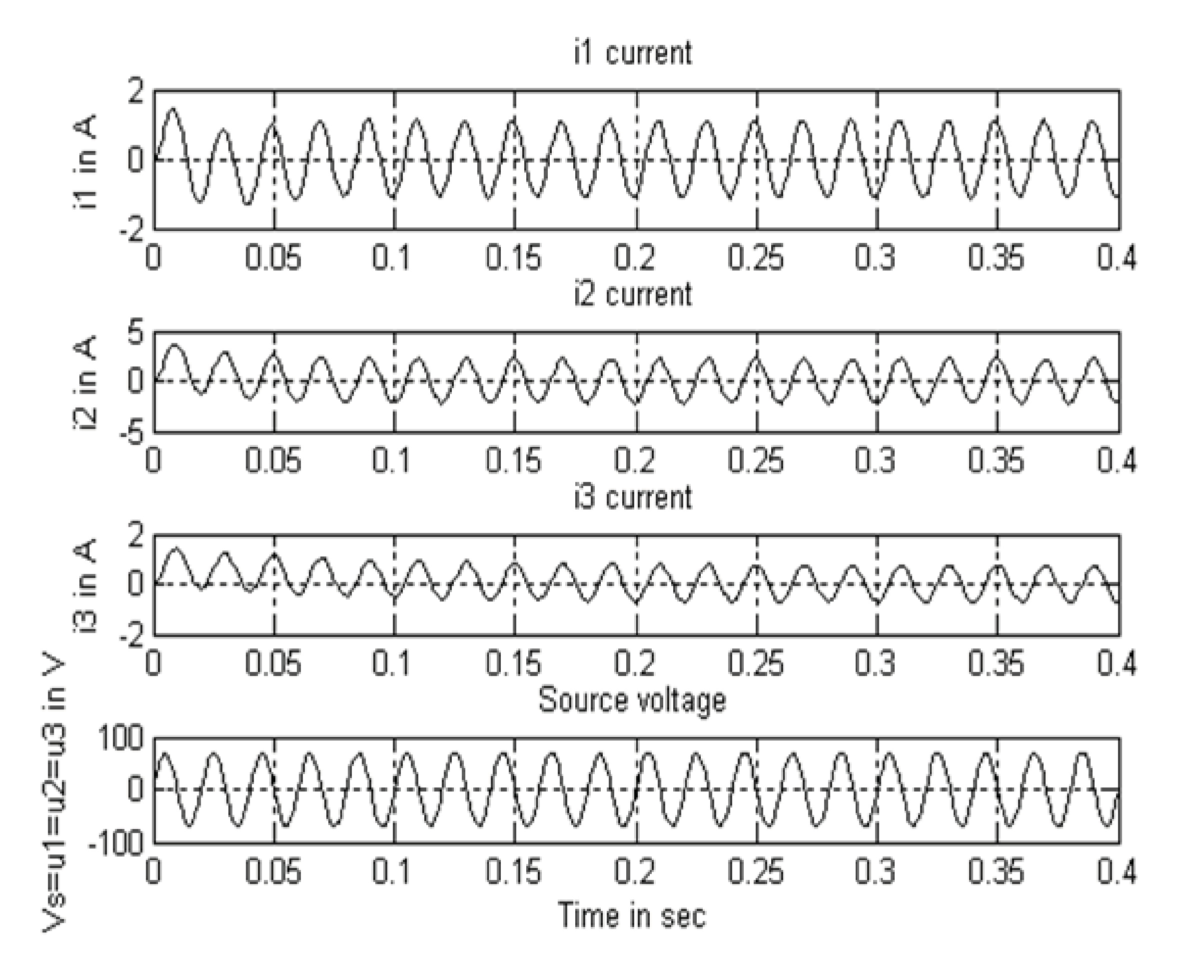


ملف المعطيات :

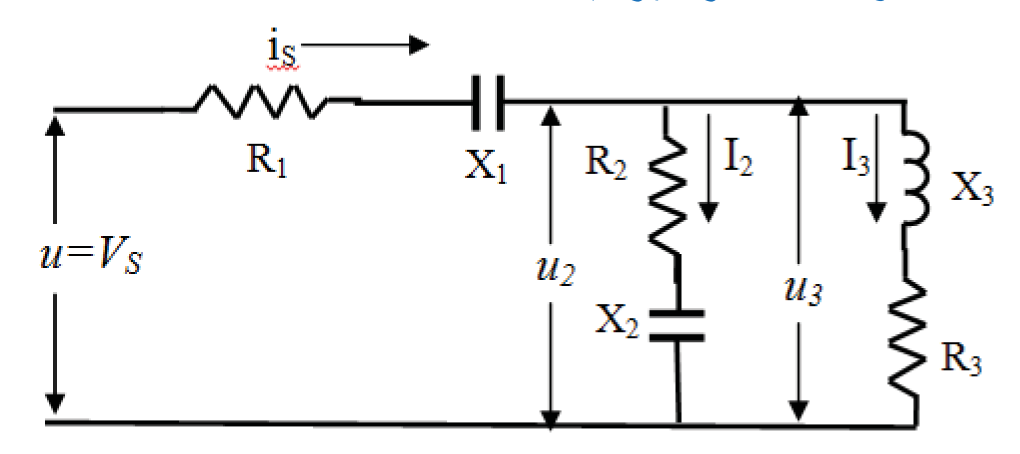
% M-file for parallel R L – RLC circuit simulation

```
% input parameters and initial conditions
% and plot results of simulation
R1 = 20; %R1 = 20 ohms
R2 = 4 ; %R2 = 4 ohms
R3 = 4 ; %R3 = 4 ohms
L1 = 0.2 ; % L1 = 0.2 Henry
L2= 0.1 ; % L2= 0.1 Henry
L3= 0.3; % L3= 0.3 Henry
C = 1000e-6 ;  % C = 1000 ?f
Vac_mag=70 ; %Volts  AC
K1= 1/ L1 ; K2 =1 / C ; K3 = R1 ; K4 =1 /L2 ; K5=R2 ; K6=1/L3
; K7=R3 ;
iL1o = 0 ; % initial value of capacitor voltage
iL2o = 0 ; % initial value of inductor current
iL3o = 0 ; % initial value of inductor current
vCo = 0 ; % initial value of capacitor voltage
tstop = 0.4; % stop time for simulation
disp('run simulation, type "return" when ready to return')
keyboard
subplot(4,1,1)
plot(simout(:,1),simout(:,2),'k')
title('i1 current')
ylabel('i1 in A')
grid
subplot(4,1,2)
plot(simout(:,1),simout(:,3),'k')
title('i2 current')
ylabel('i2 in A')
grid
subplot(4,1,3)
plot(simout(:,1),simout(:,4),'k')
title('i3 current')
ylabel('is in A')
grid
subplot(4,1,4)
plot(simout(:,1),simout(:,5),'k')
title('Source voltage')
ylabel('Vs=u1=u2=u3 in V')
xlabel('Time in sec')
grid
```

```
phiurad=0
phi1rad= phiurad - atan(((we*L1)-(1/(we*C1)))/R1)
phi1deg=( phi1rad*180)/pi
phi2rad= phiurad - atan((we*L2)/R2)
phi2deg=( phi2rad*180)/pi
phi3rad= phiurad - atan((we*L3)/R3)
phi3deg=( phi3rad*180)/pi
```



٤,١٣ النمذجة و المحاكاة لدارة كهربائية مختلطة



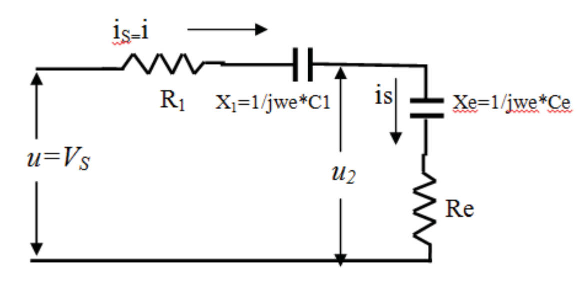
$R_1 = 5\,\Omega$, $R_2=3\,\Omega$, $R_3=4\,\Omega$, $X_1=4\,\Omega$, $X_2=4\,\Omega$, $X_3 = 3\,\Omega$
$F=50\,Hz$, $W=2\Pi f$, $U_{eff}=130/\sqrt{2}V$, $u=U\sin(wt)$

يتم وضع المخطط الصندوقي لمحاكاة للدارة المبينة على ثلاثة مراحل :

١- المرحلة الأولى : سوف نقوم بتبسيط الدارة إلى دارة مكافئة أبسط وذلك عن طريق تحصيل الفرعين (X_1 ,R_1) و (X_2 , R_2) وإيجاد محصلتهما $Ze = Re - jXe = 3.5-j0.5$ ذات الطبيعة السعوية بحيث تصبح الدارة المكافئة من الشكل التالي:

$$i_S = C_{eq}\frac{d}{dt}(V_S - R_{eq} i_S)$$

$$C_{eq}=(C_1 Ce)/(Ce+C_1) \quad , \quad R_{eq}=R_1+R_e$$



$Re=3.5\,\Omega$, $Xe=0.5\,\Omega$, $X_1= 2\,\Omega$, $R_1=2\,\Omega$
$F=50\,Hz$, $W=2\Pi f$, $U_{eff}=130/\sqrt{}\,V$, $u=U\sin(wt)$

٢- المرحلة الثانية : حساب التوتر المطبق على الجزء التفرعي

واعتباره منبع تغذية لهذا الجزء من الدارة $u_3 = u_2$.

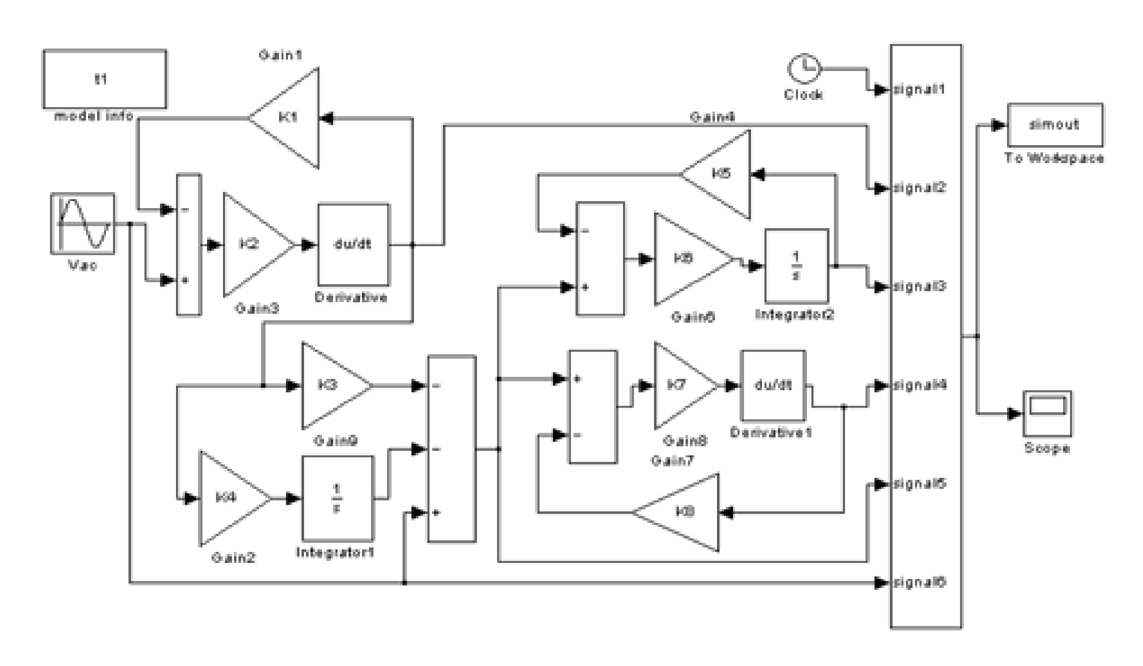
$$V_S = R_1 i_S + \frac{1}{C_1}\int i_S dt + u_2$$

$$u_2 = V_S - R_1 i_S - \frac{1}{C_1}\int i_S dt$$

3- المرحلة الثالثة : حساب التيارات الفرعية نعود إلى الدارة التفرعية واعتبار أن المنبع المغذي لها هو التوتر $u_3 = u_2$ ومنه يصبح النموذج الرياضي لهذا الجزء من الدارة:

$$i_3 = \int (\frac{u_3}{L_3} - \frac{R_3}{L_3} i_3)\, dt$$

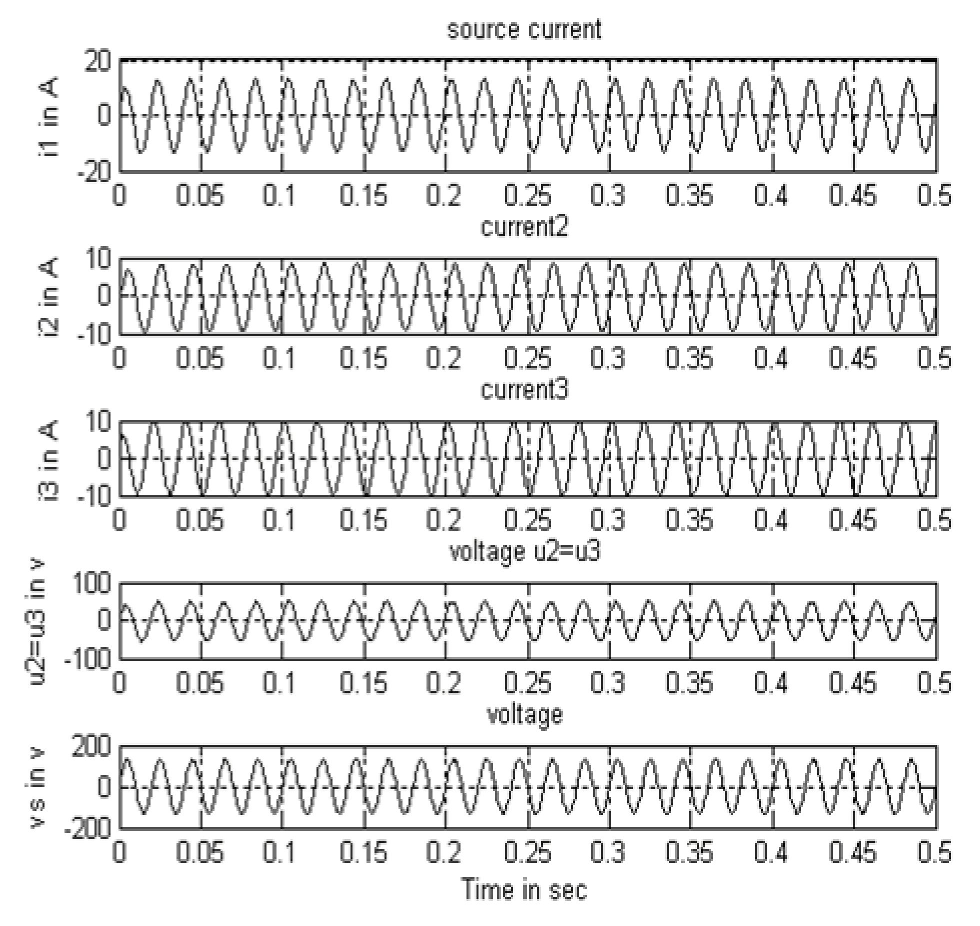
$$i_2 = C_2 \frac{d}{dt}(u_2 - R_2 i_2)$$




```
% M-file for sires  RC+ par.LC+LR circuit simulation
% input parameters and initial conditions
% and plot results of simulation
R1=5; X1=4;we=314;C1=1/(X1*we);R2=3; X2=4;
C2=1/(X2*we); X3=3; L3=X3/we; R3=4;
Z2=R2-X2*i
Z3=R3+X3*i
Z=(Z3*Z2)/(Z3+Z2)
Re=real(Z)
Req=R1+real(Z)
Xe=imag(Z)
Ce=-1/(imag(Z)*we)
Ze=sqrt((real(Z))^2+(imag(Z))^2)
Vac_mag=130;
K1=R1+real(Z)  ;
```

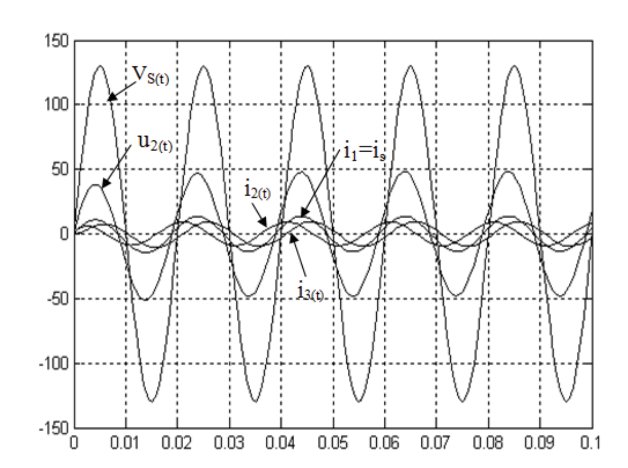
```
K2 = C1*Ce/(C1+Ce) ; %Ceq =K2= C1*Ce/(C1+Ce)
K3 =R1  ; K4=1/C1  ; K5=R3; K6=1/L3  ; K7=C2  ; K8=R2  ;
vCo = 0 ; % initial value of capacitor voltage
iLo = 0 ; % initial value of inductor current
tstop = 0.5; % stop time for simulation
disp('run simulation, type "return" when ready to return')
keyboard
subplot(5,1,1)
plot(simout(:,1),simout(:,2),'k')
title('source current')
ylabel('i1 in A')
grid
subplot(5,1,2)
plot(simout(:,1),simout(:,3),'k')
title('current2')
ylabel('i2 in A')
grid
subplot(5,1,3)
plot(simout(:,1),simout(:,4),'k')
title('Current3')
ylabel('i3 in A')
grid
subplot(5,1,4)
plot(simout(:,1),simout(:,5),'k')
title('voltage u2=u3')
ylabel('u2=u3 in v')
grid
subplot(5,1,5)
plot(simout(:,1),simout(:,6),'k')
title('voltage')
ylabel('vs in v')
xlabel('Time in sec')
grid

phitotaldeg=0-atan((imag(Ze1)-Xo)/(Ro+real(Ze1)))*180/pi
phizedeg=atan(imag(Ze1)/real(Ze1))*180/pi
phiu1deg=phitotaldeg+phizedeg
phi1deg=phiu1deg-(atan(imag(Z1)/real(Z1))*180)/pi
phi2deg=phiu1deg-(atan(imag(Z2)/real(Z2))*180)/pi
```



إظهار النتائج باستخدام العنصر simout بالأمر التالي:

```
>>
plot(simout(:,1),simout(:,2),simout(:,1),simout(:,3),simout(:,1),si
mout(:,4),simout(:,1),simout(:,5),simout(:,1),simout(:,6)),grid
```



phitotaldeg=22.3801,phizedeg=8.1301
phiu2deg=14.2500 , phi2deg=67.3801,phi3deg=-22.6199



نمذجة و محاكاة الأنظمة الديناميكية

- الأسس النظرية لمحاكاة الأنظمة الديناميكية.
- نمذجة و محاكاة المحولات الكهربائية
- نمذجة و محاكاة محركات التيار المتناوب .

النمذجة و المحاكاة للآلات الكهربائية

النمذجة هي وضع النموذج الرياضي للنظام المدروس.
المحاكاة هي وضع المخطط الصندوقي المقابل للنموذج الرياضي للنظام المدروس.

١,١٤ التحويلات ثلاثية الطور

في دراسة النظم الكهربائية ، فإن التحويلات الرياضية غالباً ما تستخدم من أجل فك الترابط بين المتغيرات ، وذلك من أجل تسهيل حل المعادلات المعقدة المتغيرة مع الزمن أو الانتقال إلى أساس مرجعي لكافة المتغيرات.
فلو أخذنا نظرية المركبات المتناظرة كمثال سنلاحظ أن التحويل العقدي يستخدم لتفكيك الترابط بين الأطوار (a,b,c) والحصول على المركبات :

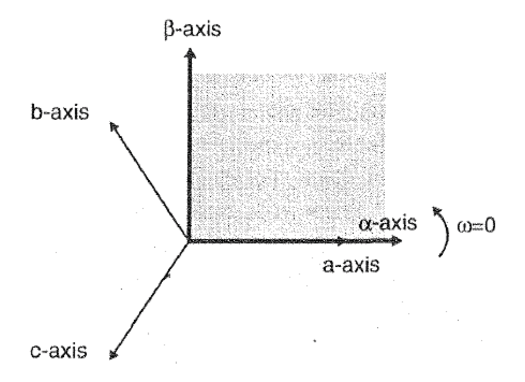
$$[f_{012}] = [T_{012}][f_{abc}]$$

حيث f قد تكون تيار أو توتر أو سيالة ..... إلخ
ومصفوفة التحويل تعطى :

$$[T_{012}] = \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 1 & a^2 & a \\ 1 & a & a^2 \end{bmatrix} \qquad \text{حيث أن } a = e^{j\frac{2\pi}{3}}$$

١- تحويل كلارك

إن الطورين الثابتين (المستقرين) في تحويل كلارك يتغيران ويرمز لهما بالشكل بالرمزين (α, β).



الشكل المبين يوضح المحاور (a,b,c) وعلاقتها مع المحورين (α, β) حيث نلاحظة انطباق المحور α مع الطور الأول ولكن المحور β يتأخر عنه بزاوية ٩٠ أما المركبة الثالثة فتعرف بالمركبة الصفرية

$$[f_{\alpha\beta 0}] = [T_{\alpha\beta 0}][f_{abc}]$$

$$[T_{\alpha\beta0}] = \frac{2}{3}\begin{bmatrix} 1 & \frac{-1}{2} & \frac{-1}{2} \\ 0 & \sqrt{\frac{3}{2}} & -\sqrt{\frac{3}{2}} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad \& \quad [T_{\alpha\beta0}]^{-1} = \begin{bmatrix} 1 & 0 & 1 \\ \frac{-1}{2} & \sqrt{\frac{3}{2}} & 1 \\ \frac{-1}{2} & -\sqrt{\frac{3}{2}} & 1 \end{bmatrix}$$

٢- تحويل بارك

تحويل بارك معروف كثيراً في التحويل من نظام الأطوار الثلاثة إلى نظام الطورين في تحليل الآلات التزامنية ، ومعادلة التحويل تعطى بالصيغة :

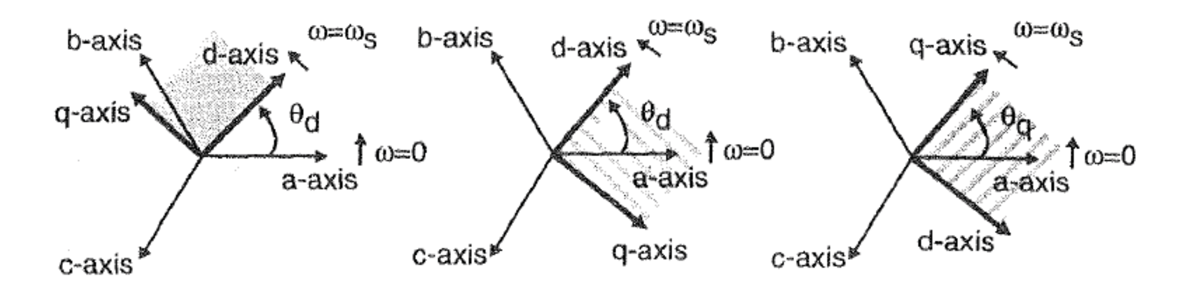
$$[f_{dq0}] = [T_{dq0}(\theta_d)][f_{abc}]$$

$$[T_{dq0}(\theta_d)] = \frac{2}{3}\begin{bmatrix} \cos(\theta_d) & \cos(\theta_d - \frac{2\pi}{3}) & \cos(\theta_d + \frac{2\pi}{3}) \\ -\sin(\theta_d) & -\sin(\theta_d - \frac{2\pi}{3}) & -\sin(\theta_d + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

$$[T_{dq0}(\theta_d)]^{-1} = \begin{bmatrix} \cos(\theta_d) & -\sin(\theta_d) & 1 \\ \cos(\theta_d - \frac{2\pi}{3}) & -\sin(\theta_d - \frac{2\pi}{3}) & 1 \\ \cos(\theta_d + \frac{2\pi}{3}) & -\sin(\theta_d + \frac{2\pi}{3}) & 1 \end{bmatrix}$$

حيث يستخدم تحويل بارك من الانتقال بالكميات الثابتة للآلة التزامنية من المحاور ثلاثية الطور إلى نظام المحورين (d,q) والثابتة نسبة للدائر.

بعض المراجع تعتبر المحور q متأخر عن المحور d بزاوية ٩٠ كما وضحنا ، وهناك أخرين يعتبرون أن المحور q متقدم عن المحور d بزاوية ٩٠ ويمكن التعامل م التحويل كتابع للزاوية θ_q والشكل التالي يوضح ذلك



$$[f_{dq0}] = [T_{dq0}(\theta_q)][f_{abc}]$$

$$[T_{dq0}(\theta_q)] = \frac{2}{3}\begin{bmatrix} \cos(\theta_q) & \cos(\theta_q - \frac{2\pi}{3}) & \cos(\theta_q + \frac{2\pi}{3}) \\ \sin(\theta_q) & \sin(\theta_q - \frac{2\pi}{3}) & \sin(\theta_q + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

$$[T_{dq0}(\theta_q)]^{-1} = \begin{bmatrix} \cos(\theta_q) & \sin(\theta_q) & 1 \\ \cos(\theta_q - \frac{2\pi}{3}) & \sin(\theta_q - \frac{2\pi}{3}) & 1 \\ \cos(\theta_q + \frac{2\pi}{3}) & \sin(\theta_q + \frac{2\pi}{3}) & 1 \end{bmatrix}$$

إن العلاقة بين (θ_q, θ_f) كما تعرف في تحويل بارك هي : $\theta_q = \theta_d + \frac{\pi}{2}$

وبالتالي :

$$\cos(\theta_d + \frac{\pi}{2}) = -\sin(\theta_d)$$

$$\sin(\theta_d + \frac{\pi}{2}) = \cos(\theta_d)$$

وبالتالي نلاحظ أن التحويلان $[T_{dq0}(\theta_q)]$ و $[T_{dq0}(\theta_d)]$ هما متشابهين تماماً ماعدا ترتيب المحاور (d,q).

مثال :

ارسم المخطط الصندوقي المقابل للمعادلات التالية :

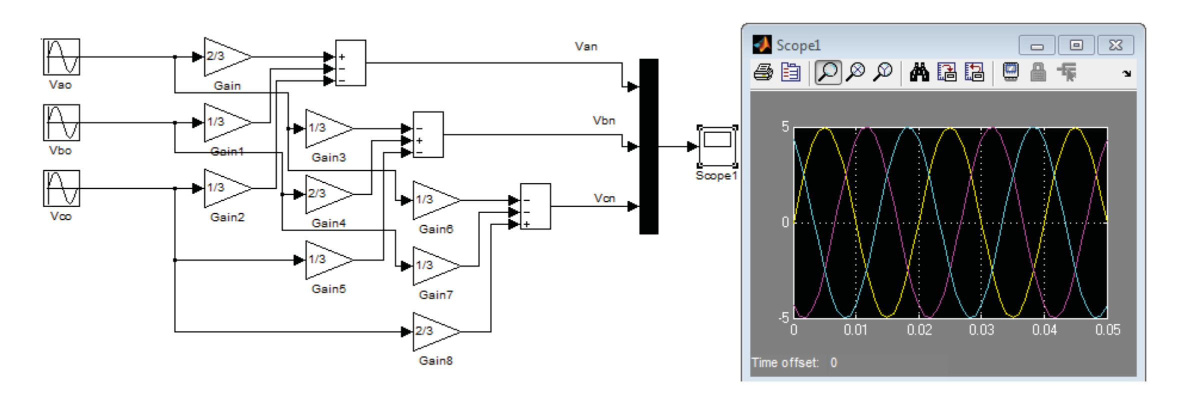
$$\begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & \frac{-1}{3} & \frac{-1}{3} \\ \frac{-1}{3} & \frac{2}{3} & \frac{-1}{3} \\ \frac{-1}{3} & \frac{-1}{3} & \frac{2}{3} \end{bmatrix}\begin{bmatrix} V_{ao} \\ V_{bo} \\ V_{co} \end{bmatrix}$$

<u>طريقة أولى :</u>

نحتاج في هذه الطريقة إلى :

- منبع جيبي متناوب.
- عناصر تضخيم (Gain)
- مازج (Mux)
- عنصر إظهار للنتائج (Scope)

والشكل التالي يبين المخطط الصندوقي (Amplitude = 5 – f=50Hz – t=0.05)

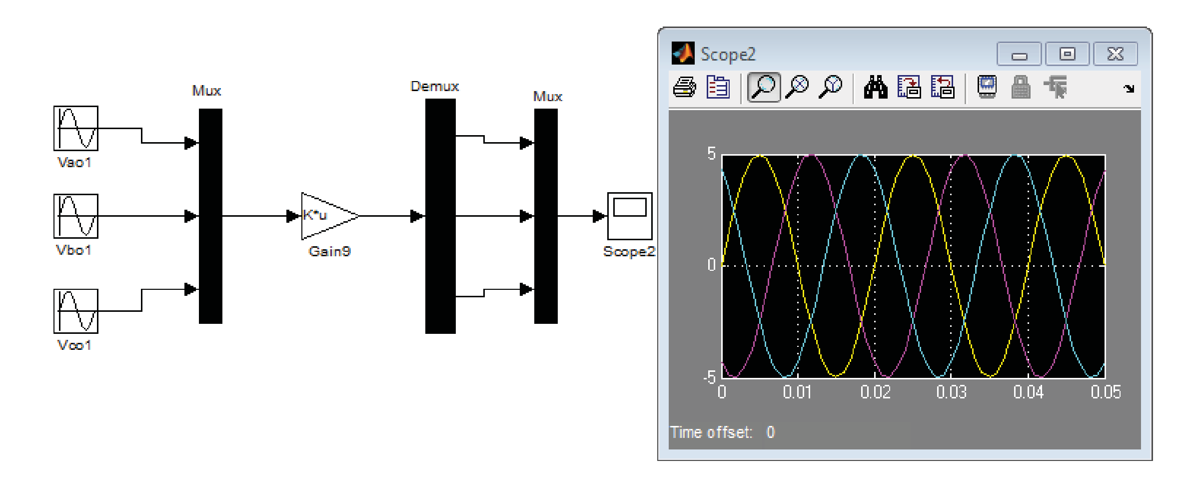


طريقة ثانية :

- منبع جيبي متناوب.
- مازج Mux و Demux (عكس المازج)
- عنصر تكبير (Gain) ولكن هنا نختار من خصائصه من

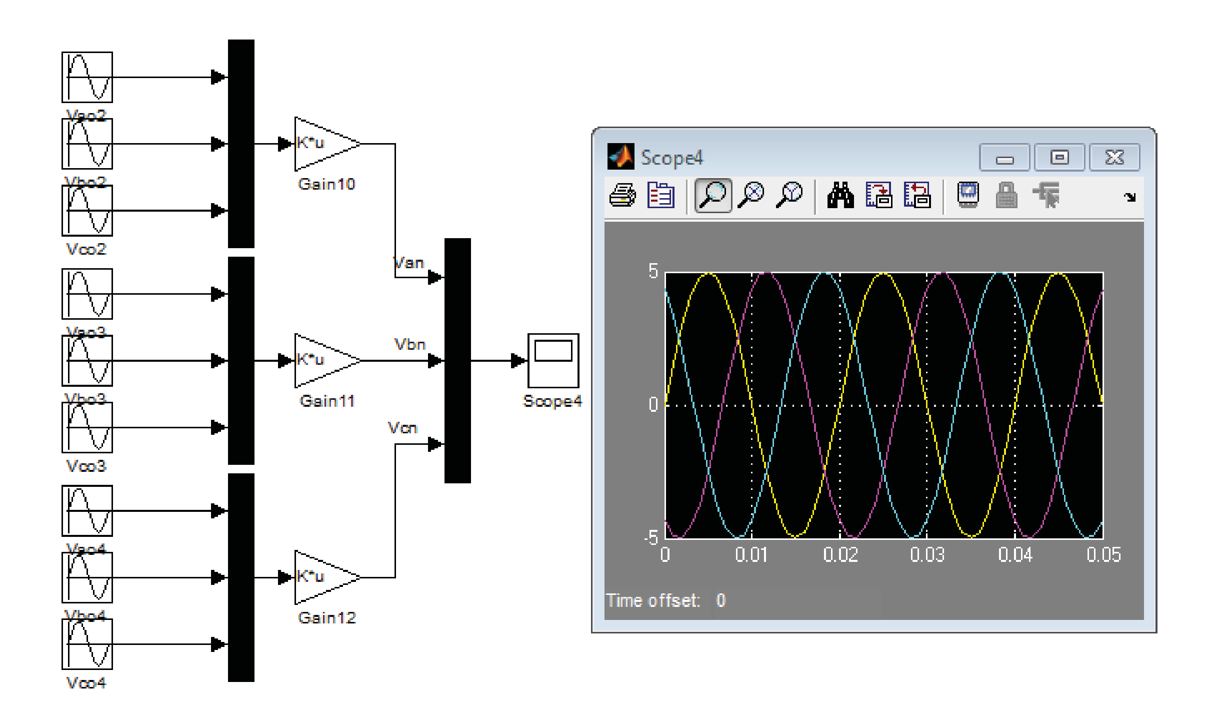
Multiplicatioc → Matrix(k*u)

حيث k مصفوفة ثابتة بدلاً من ثابت (عنصر)

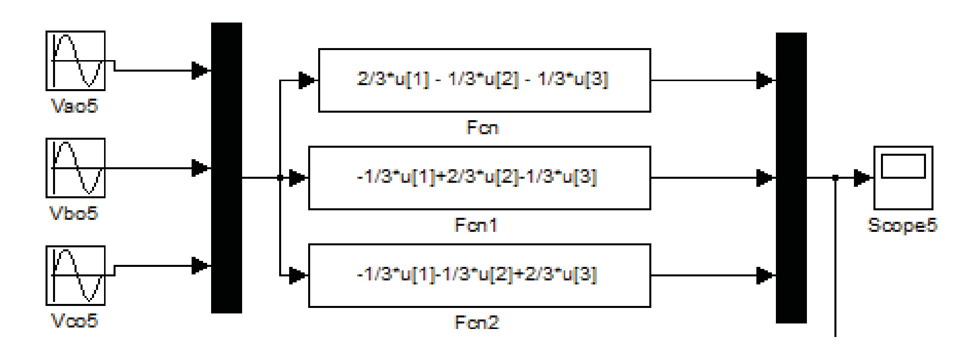


طريقة ثالثة :

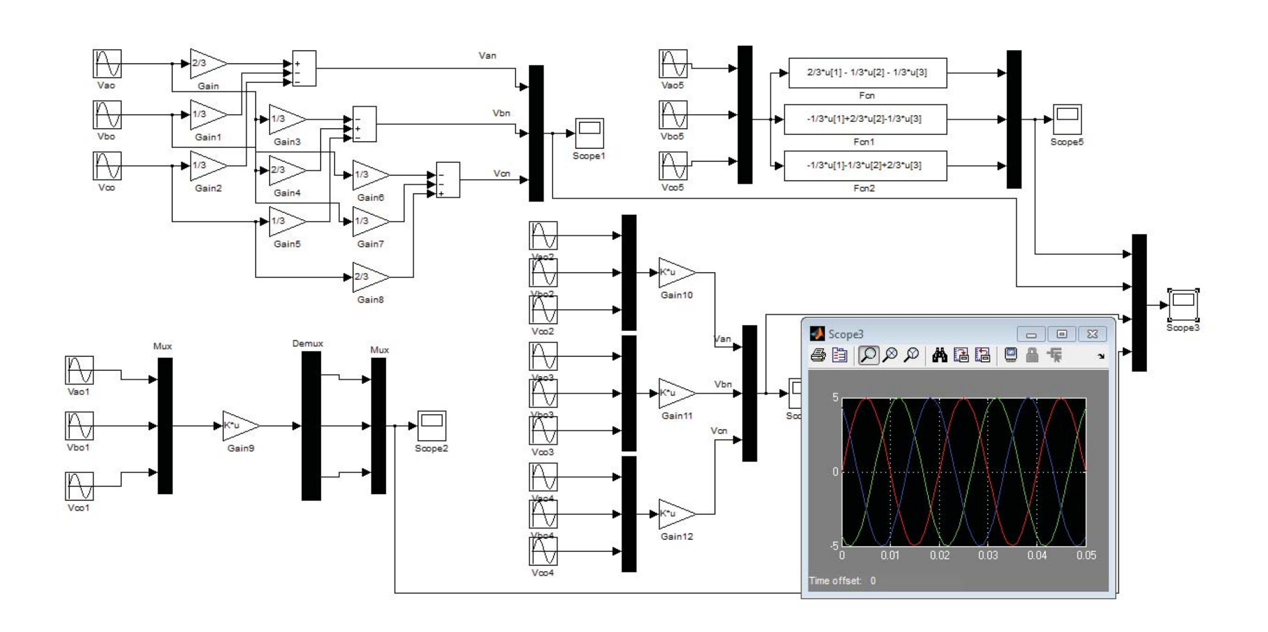
ملاحظة : لابد في هذه الطريقة من الانتباه أن المصفوفة المدخلة في عناصر التضخيم مختلفة عن الطريقة السابقة وذلك لأنه يتم التعامل مع كل طور على حدى مع المحاكاة.



طريقة رابعة :



من أجل التأكد من أن النتائج متطابقة تماماً بكل الطريق ما علينا إلا دمج نتائج الحالات الأربعة باستخدام عنصر (Mux) ومن ثم إظهارها على راسم وفي حال تطابق جميع النتائج فلن يظهر لنا إلا ثلاثة أشكال مزاحة عن بعضها بزاوية ١٢٠ ، والشكل التالي يوضح ذلك :

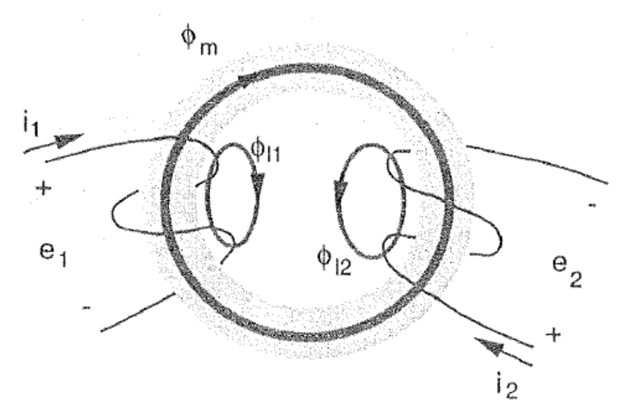


٢,١٤ نمذجة ومحاكاة المحولات الكهربائية

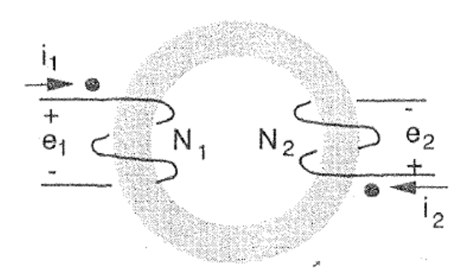
سنبدأ باستنتاج معادلات السيالة المغناطيسية (التشابك المغناطيسي) والتوتر على أطراف المحولة ذات الملفين (ابتدائي وثانوي) ، آخذين بعين الاعتبار المقاومة والسيالة التسربية للملفات والممانعة المغناطيسية للنواة ، ومن ثم سنستنتج الدارة المكافئة التي تمثل المحولة.

١,٢,١٤ معادلات السيالة والتشابك في المغناطيسي بين الملفات في المحول

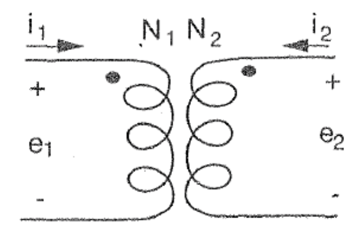
تتكون السيالة المغناطيسية في كل ملف من ملفات المحولة من جزأين إذا ما تم أخذ السيالة التسربية بعين الاعتبار ، كما يبين الشكل



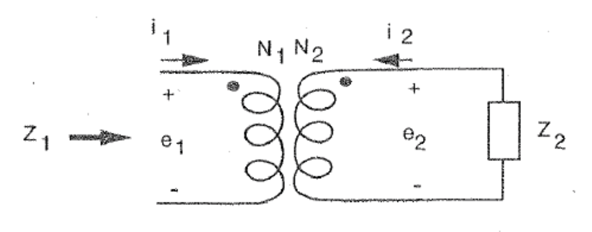
الشكل الموضح يبين مسارات الفيوض في المحول أحادي الطور والشكل التالي يوضح المخطط التمثيلي للمحول أحادي الطور.



(a) Windings and core of a two-winding transformer



(b) Circuit symbol of a two-winding transformer



(c) Referred value of Z2

حيث تتكون السيالة الكلية من مركبتين حيث المركبة الأولى تمثل السيالة المتبادلة وهي متساوية لكلا الملفين (هي السيالة التي تربط بين الملفين) والمركبة الثانية تمثل السيالة التسربية الخاصة بكل ملف.

$$\emptyset_2 = \emptyset_{L2} + \emptyset_m \quad - \quad \emptyset_1 = \emptyset_{L1} + \emptyset_m$$

حيث :

$\emptyset_{12} = \emptyset_{21}$: تمثلان السيالة المتبادلة ، يرمز لها أيضاً $\emptyset_m$

$\emptyset_{L2} = \emptyset_{L1}$: السيالتان التسربيتان للملفين الابتدائي والثانوي على التوالي.

ينشأ الفيض المتبادل في المحول المثالي من القوة المحركة المغناطيسية للملفين والذي يدور بنفس المسار في النواة. بفرض أن N_1 هو عدد لفات الملف الأول (الابتدائي) ، يربط فعلياً كل من الفيض المتبادل والفيض المتسرب ، فإن التشابك المغناطيسي في الملف الأول (الابتدائي) يحدد بأنه عدد لفات الملف الابتدائي مضروباً بالتدفق الكلي الناتج عن الملف الأول :

$$\lambda_1 = N_1 . \emptyset_1 = N_1(\emptyset_{L1} + \emptyset_m)$$

ويمكن ان يعبر عن الطرف الأيمن للمعادلة الأخيرة بدلالة تيارات الملفات ، باستبدال الفيض المتبادل والفيض المتسرب بالقوى المحركة المغناطيسية والمسايرات الخاصة بهذه الملفات. الفيض المتسرب في الملف الأول تسببه القوى المحركة المغناطيسية الناتجة عن الملف الأول والفيض المتبادل يسببه مجموع القوى المحركة المغناطيسية ، وبالتعويض عن الفيض المتبادل والفيض التسربي بالمعادلة الأخيرة نجد :

$$\lambda_1 = N_1.\emptyset_1 = N_1(\emptyset_{L1}+\emptyset_m) = L_{11}.i_1 + L_{12}.i_2$$

$$\lambda_2 = N_2.\emptyset_2 = N_2(\emptyset_{L2}+\emptyset_m) = L_{21}.i_1 + L_{22}.i_2$$

$$L_{11} = L_{l1} + L_{m1}$$

$$L_{12} = \frac{N_2}{N_1} L_m i_2$$

حيث :

$L_{11} - L_{22}$: المحارضات الذاتية للملفين الأولي والثانوي على التوالي.

$L_{12} - L_{21}$: المحارضات المتبادلة بين الملفين الأولي والثانوي على التوالي.

٢,٢,١٤ معادلات التوتر

إن التوتر المتحرض في كل ملف يساوي إلى معدل تغير التشابك المغناطيسي ولذلك يمكن أن يعطى التوتر المتحرض في الملف الأول (الابتدائي) بالعلاقة التالية :

$$e_1 = \frac{d\lambda_1}{dt} = L_1 \frac{di_1}{dt} + L_{12} \frac{di_2}{dt} = L_{l1} \frac{di_1}{dt} + L_{m1} \frac{d}{dx} (i_1 + i_2')$$

$$e_2' = L_{12}' \frac{di_2'}{dt} + L_{m1} \frac{d}{dx} (i_1 + i_2')$$

ومعادلات التوتر تكون كالتالي :

$$v_1 = r_1.i_1 + e_1 = r_1.i_1 + L_{l1} \frac{di_1}{dt} + L_{m1} \frac{d}{dx} (i_1 + i_2')$$

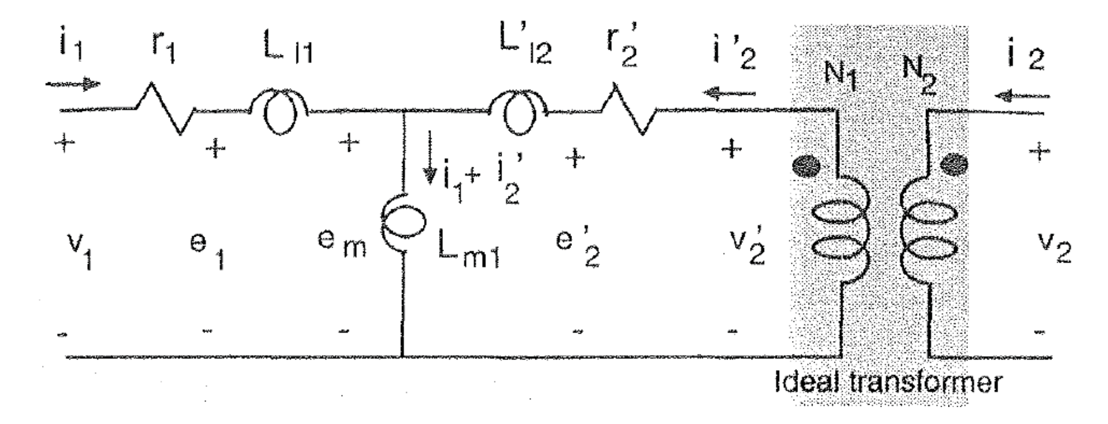
$$v_2' = i_2'.r_2' + L_{12}' \frac{di_2'}{dt} + L_{m1} \frac{d}{dx} (i_1 + i_2')$$

٣,٢,١٤ الدارة المكافئة لمحولة أحادية الطور

إن شكل معادلات التوتر المبين يحدد شكل الدارة المكافئة لمحولة ثنائية الملفات وذلك كما يبين الشكل التالي ، إن إشارة الفتحة (′) تدل أن قيم الثانوي منسوبة إلى الطرف الأولى (الابتدائي) .

إن قيم محددات الدارة للملف الثانوي منسوبة للأولي تحدد بالعلاقة :

$$r_2' = (\frac{N_1}{N_2})^2.r_2 \quad - \quad L_{12}' = (\frac{N_1}{N_2})^2.L_{12}$$



٤,٢,١٤ نمذجة المحولة أحادية الطور

من أجل نمذجة المحولة أحادية الطور (معرفة تغييرات مجاهيل بارمترات المحولة بالاعتماد على القيم المعلومة فيها) نحتاج إلى المعادلات التالية :

$$i_1 = \frac{\psi_1 - \psi_m}{X_{l1}} \quad ..... 1$$

$$i_2' = \frac{\psi_2' - \psi_m}{X_{l2}'} \quad ..... 2$$

$$\frac{1}{X_M} = \frac{1}{x_{m1}} + \frac{1}{x_{l1}} + \frac{1}{x_{l2}'}$$

$$\psi_m = x_M \left(\frac{\psi_1}{x_{l1}} + \frac{\psi_2'}{x_{12}'} \right) \quad ..... 3$$

$$\psi_1 = \int \{ w_b . v_1 - w_b . r_1 . i_1 \} dt \quad ..... 4$$

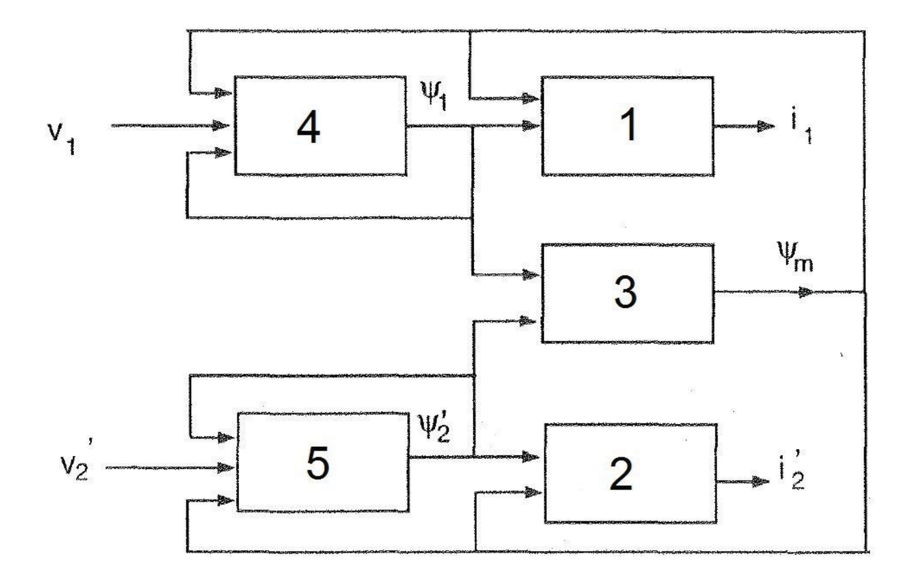
$$\psi_2' = \int \{ w_b . v_2' - w_b . r_2' . i_2' \} dt \quad ..... 5$$

نستنتج أن مجموع المعادلات الأخيرة تشكل النموذج الديناميكي الأساسي لمحول ذي ملفين أولي وثانوي وهذه المعادلات غير مستقلة خطياً من حيث التشابك المغناطيسي ويمكن إضافة المفاقيد الحديدية لها إذا دعت الضرورة.

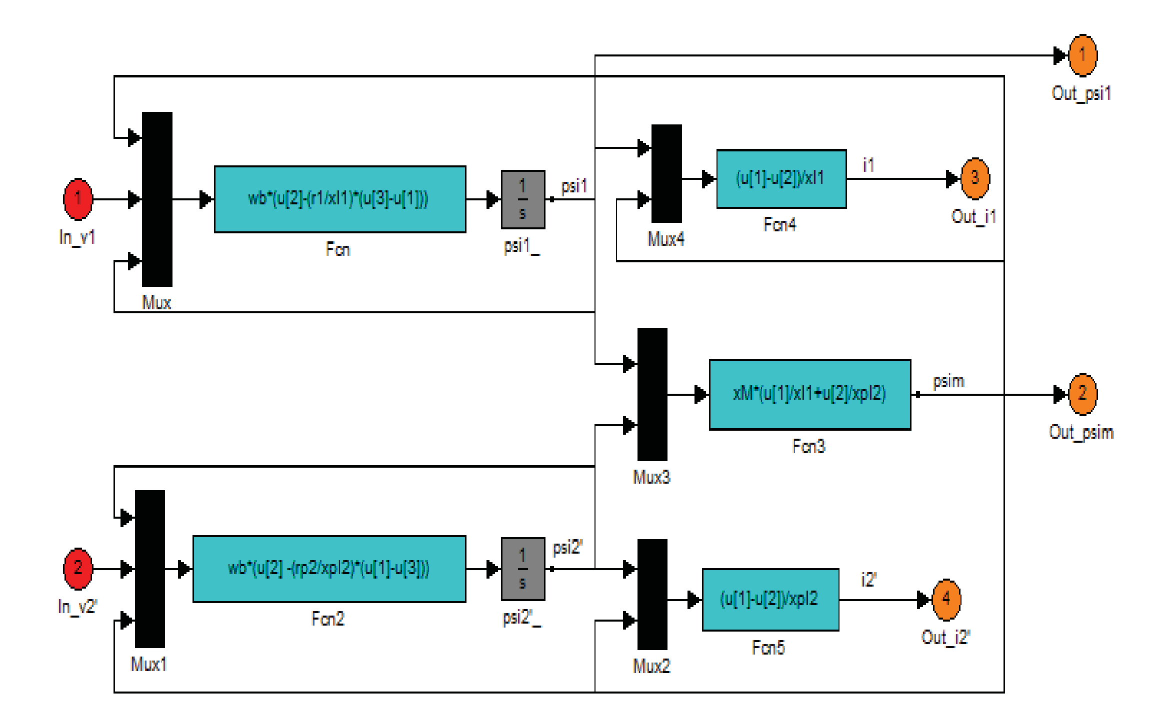
وعند تنفيذ ملف المحاكاة يجب الأخذ بعين الاعتبار لما يلي :

- التشابك المغناطيسي كمتغيرات داخلية.
- توترات الأطراف هي معطيات الدخل المطلوبة.
- تيارات الملفات معطيات الخرج الرئيسية.

والمخطط التالي يدعى بالمخطط الانسيابي لعملية النمذجة لمحول بملفين (أحادي الطور) ...

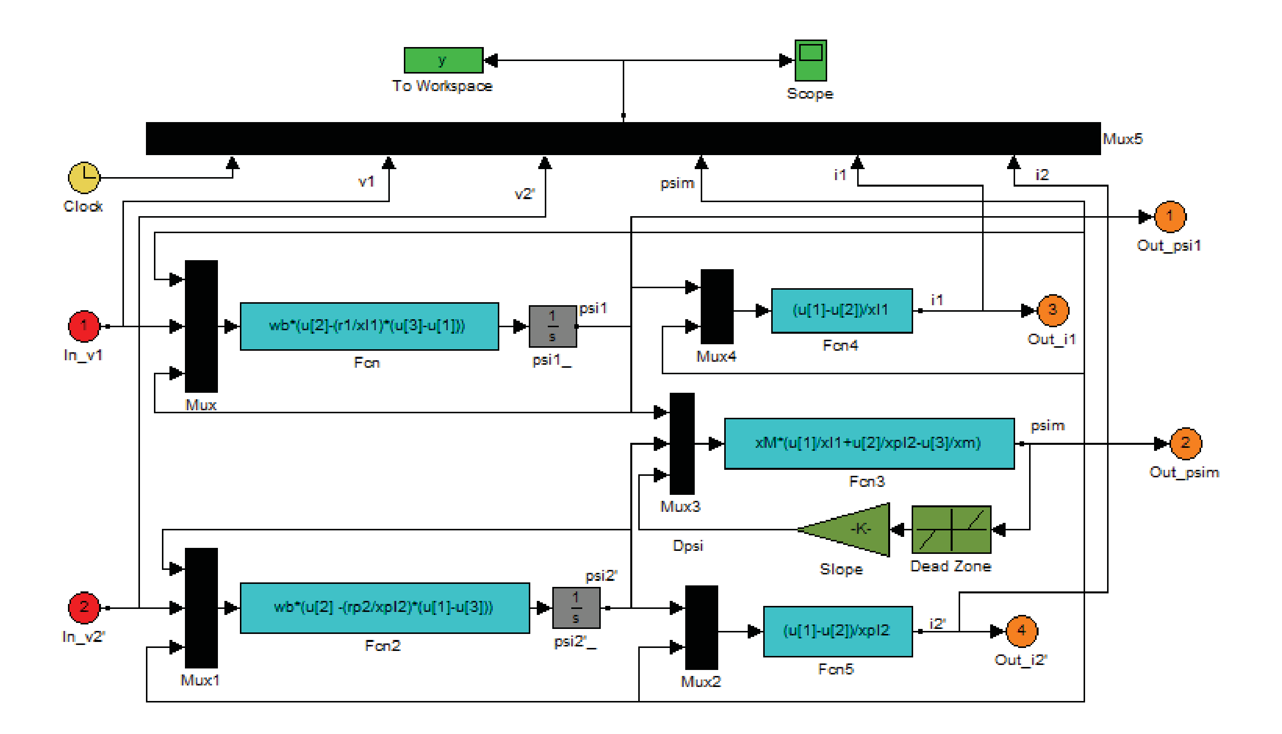


ونموذج المحاكاة المقابل لهذا المخطط مبين بالشكل التالي ...



يجب الانتباه دوماً بالقيام بربط المخطط الصندوقي مع ملف M-file لتعريف كافة المجاهيل (في المخطط) داخل الملف ، ويجب تشغيل الملف أولاً وإدخال قيم العناصر المطلوبة ومن ثم تشغيل النموذج المقابل لهذا الملف.

بالنسبة للمخطط السابق يمكن مشاهدة المنحنيات من خلال تنفيذ النموذج التالي :



مثال (١) :

في النموذج السابق رأينا الحالة العامة لمخطط المحولة أحادية الطور ، ولكن قيمة توتر الخرج يحددها الحمل وسنناقش في هذا المثال اختبارات المحولة الثلاثة (القصر – اللا حمل – التحميل) وذلك من خلال جعل المستخدم يحدد قيمة مقاومة الحمل ومن ثم يشاهد المنحنيات الناتجة عن قيمة الحمل التي قام بتحديدها ، وذلك من أجل محولة لها المواصفات التالية :

S = 1500 VA – V = 120 Volt – f = 60 Hz – N = 0.5 – r1= 0.25 – r2’= 0.134

Xl1 = 0.056 – xpl2 = 0.056

في هذا المثال لو جعلنا قيمة مقاومة الحمل مساوية للصفر سنكون قد اجرينا اختبار القصر على المحولة وستكون المنحنيات الناتجة ممثلة للمحولة أحادية الطور في حالة القصر أما لو جعلنا قيمة الحمل كبيرة جداً (100Zb) وأكبر فإن الاختبار المقابل لهذه الحالة هو اختبار اللا حمل ويمكن إجراء حالة التحميل الإفتراضية بإعطاء قيم مناسبة للحمل.

في هذا المثال لن نهمل أثر التشبع وسندخل حلقة خاصة بحالة التشبع في مخطط نمذجة المحولة وسندخل قيم تغيرات الفيض (السيالة) ضمن ملف القراءة وستتم معالجته من خلال العنصر Look-up table الذي سنراه في نموذج المحاكاة .

والبرنامج الذي يجب كتابته في ملف التحرير موضح كالتالي :

```
% محولة ذات ملفين
clear all % حذف كافة المتغيرات في الذاكرة
% تحديد بارامترات الدارة المكافئة للمحولة وكذلك منحنى المغنطة

Vrated = 120; % الجهد
Srated = 1500; % الاستطاعة الظاهرية
Frated = 60; % التردد
Zb = Vrated^2/Srated; % ممانعة الأساس
wb = 2*pi*Frated; % تردد الأساس
Vpk = Vrated*sqrt(2); القيمة العظمى للجهد
NpbyNs = 120/240; % نسبة التحويل للمحول
r1 = 0.25; % مقاومة الأولي
rp2 = 0.134; % مقاومة الثانوي منسوبة للأولي
xl1 = 0.056; % المفاعلة التسريبية للأولي
xpl2 = 0.056; % المفاعلة التسريبية للثانوي منسوب للأولي
xm = 708.8; % المفاعلة المغناطيسية
xM = 1/(1/xm + 1/xl1 + 1/xpl2);

% منحنى المغنطة المقابل  للمصفوفتين التاليتين

Dpsi=[ -2454.6 -2412.6 -2370.5 -2328.5 -2286.4 -2244.4 -2202.3 ...
-2160.3 -2118.2 -2076.1 -2034.1 -1992.0 -1950.0 -1907.9 -1865.9 ...
-1823.8 -1781.8 -1739.7 -1697.7 -1655.6 -1613.6 -1571.5 -1529.5 ...
-1487.4 -1445.3 -1403.3 -1361.2 -1319.2 -1277.1 -1235.1 -1193.0 ...
-1151.0 -1108.9 -1066.9 -1024.8 -982.76 -940.71 -898.65 -856.60 ...
-814.55 -772.49 -730.44 -688.39 -646.43 -604.66 -562.89 -521.30 ...
-479.53 -438.14 -396.75 -355.35 -313.96 -272.56 -231.17 -192.60 ...
-154.04 -116.41 -81.619 -46.822 -19.566 0.0000 0.0000 0.0000
0.0000 ...
0.0000 0.0000 19.566 46.822 81.619 116.41 154.04 192.60 231.17 ...
272.56 313.96 355.35 396.75 438.14 479.53 521.30 562.89 604.66 ...
646.43 688.39 730.44 772.49 814.55 856.60 898.65 940.71 982.76 ...
```

```
1024.8 1066.9 1108.9 1151.0 1193.0 1235.1 1277.1 1319.2 1361.2 ...
1403.3 1445.3 1487.4 1529.5 1571.5 1613.6 1655.6 1697.7 1739.7 ...
1781.8 1823.8 1865.9 1907.9 1950.0 1992.0 2034.1 2076.1 2118.2 ...
2160.3 2202.3 2244.4 2286.4 2328.5 2370.5 2412.6 2454.6 ];

psisat=[ -170.21 -169.93 -169.65 -169.36 -169.08 -168.80 -168.52 ...
-168.23 -167.95 -167.67 -167.38 -167.10 -166.82 -166.54 -166.25 ...
-165.97 -165.69 -165.40 -165.12 -164.84 -164.56 -164.27 -163.99 ...
-163.71 -163.43 -163.14 -162.86 -162.58 -162.29 -162.01 -161.73 ...
-161.45 -161.16 -160.88 -160.60 -160.32 -160.03 -159.75 -159.47 ...
-159.18 -158.90 -158.62 -158.34 -157.96 -157.39 -156.83 -156.07 ...
-155.51 -154.57 -153.62 -152.68 -151.74 -150.80 -149.85 -146.08 ...
-142.31 -137.60 -130.06 -122.52 -107.44 -84.672 -42.336 0.0000 ...
0.0000 42.336 84.672 107.44 122.52 130.06 137.60 142.31 146.08 ...
149.85 150.80 151.74 152.68 153.62 154.57 155.51 156.07 156.83 ...
157.39 157.96 158.34 158.62 158.90 159.18 159.47 159.75 160.03 ...
160.32 160.60 160.88 161.16 161.45 161.73 162.01 162.29 162.58 ...
162.86 163.14 163.43 163.71 163.99 164.27 164.56 164.84 165.12 ...
165.40 165.69 165.97 166.25 166.54 166.82 167.10 167.38 167.67 ...
167.95 168.23 168.52 168.80 169.08 169.36 169.65 169.93 170.21 ];

% تحديد بارامترات المحاكاة
tstop = 0.2; % زمن التنفيذ
Psi1o = 0; % قيم ابتدائية
Psip2o = 0; % قيم ابتدائية

repeat_run = 'Y'; % لتكرار البرنامج أكثر من مرة
while repeat_run == 'Y'

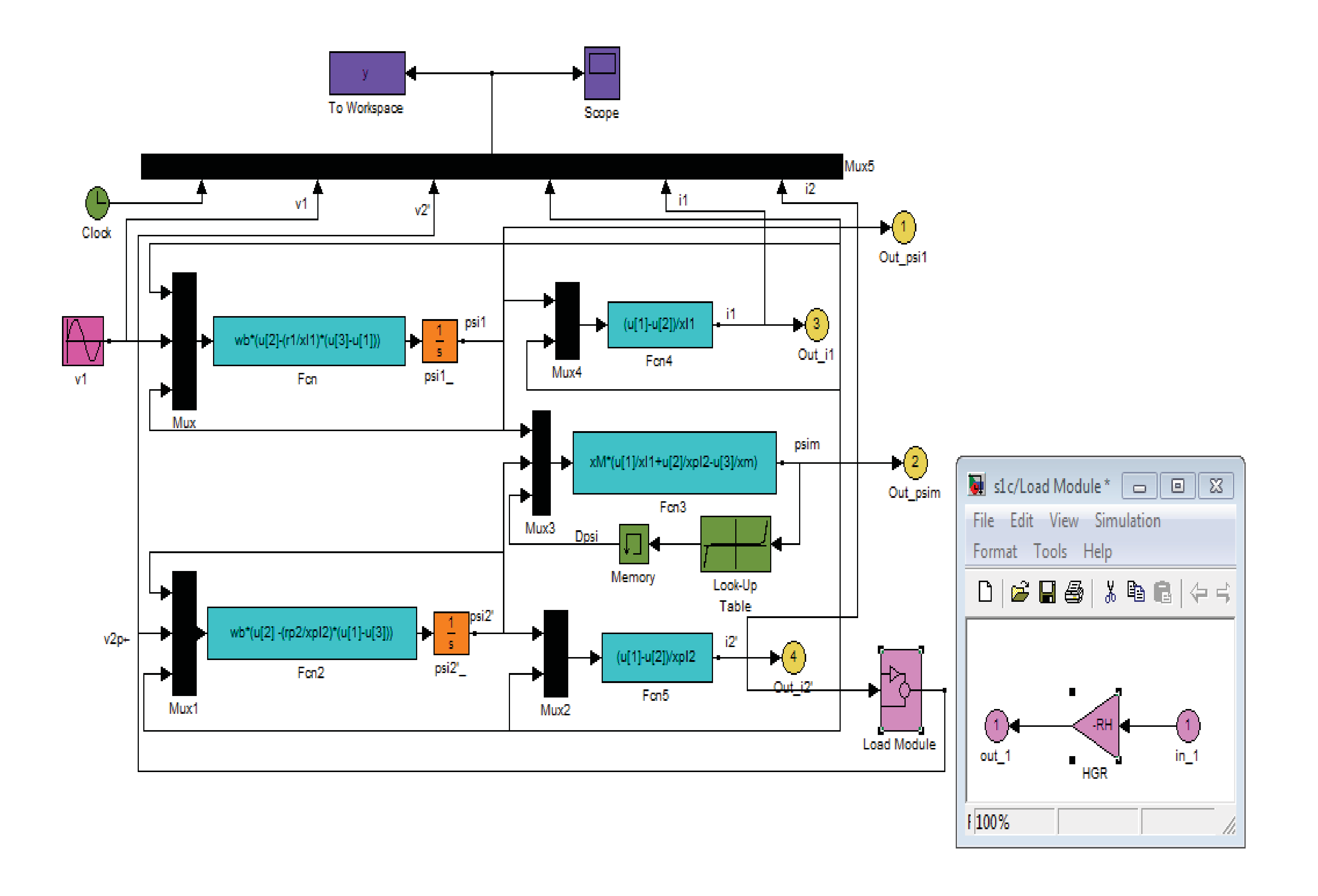
disp('Enter value of RH, the high-gain resistor to develop v2''')
RH = input('Enter ohmic value of high gain resistor :  ')
disp('')
disp('Run simulation then type ''return '' for plots')
keyboard
clf;
subplot(3,1,1)
plot(y(:,1),y(:,2),'-')
```

```
ylabel('v1 in V')
title('primary voltage')
subplot(3,1,2)
plot(y(:,1),y(:,3),'-')
ylabel('v2'' in V')
title('secondary voltage')
subplot(3,1,3)
plot(y(:,1),y(:,4),'-')
ylabel('psim in Wb/sec')
title('mutual flux')

h2=figure;
clf;
subplot(3,1,1)
plot(y(:,1),y(:,5),'-')
ylabel('i1 in A')
title('primary current')
subplot(3,1,2)
plot(y(:,1),y(:,6),'-')
ylabel('i2'' in A')
xlabel('Time in sec')
title('secondary current')

disp('Save plots before typing return')
keyboard
close (h2)

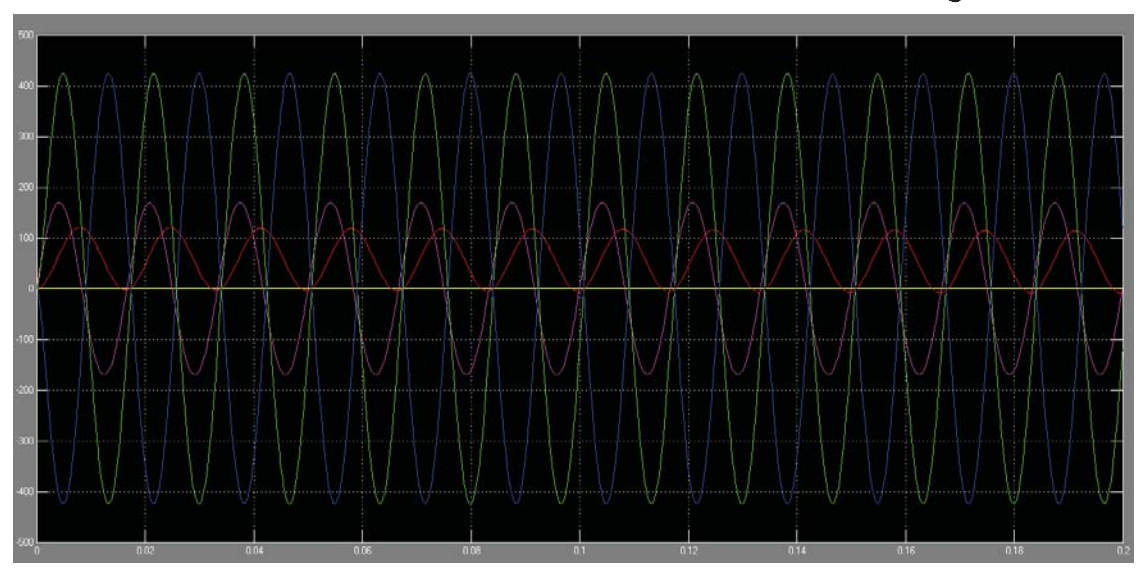
% تكرار البرنامج
repeat_run =  input('Repeat with new system condition? Y/N: ','s');
if isempty(repeat_run) % رفض التكرار
repeat_run = 'N';
end
end % إنهاء الحلقة
```

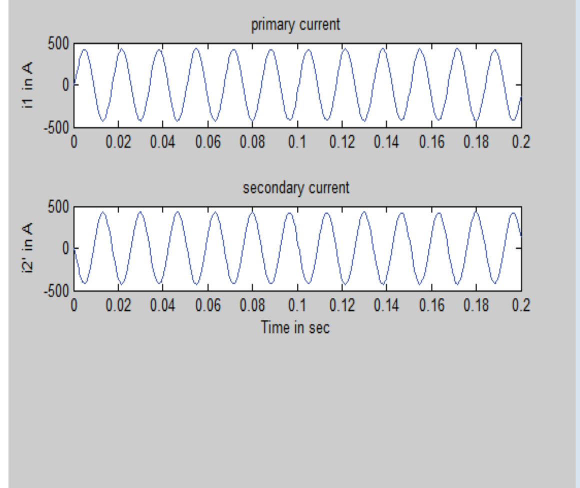


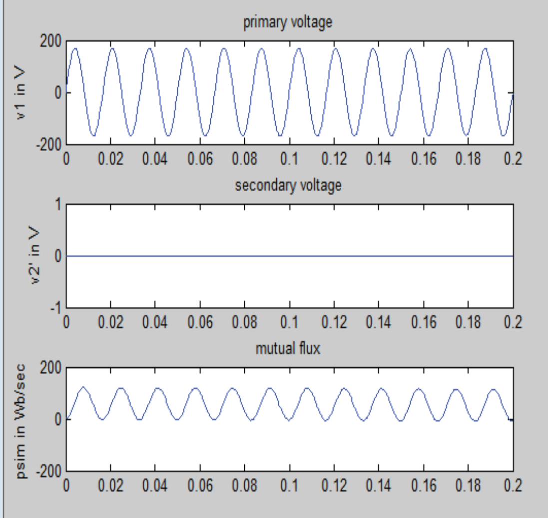
To Workspace
y
Scope
Mux5
Clock
v1
v2'
i1
i2
Out_psi1
v1
Mux
wb*(u[2]-(r1/xl1)*(u[3]-u[1]))
Fcn
psi1
psi1_
Mux4
(u[1]-u[2])/xl1
Fcn4
i1
Out_i1
Mux3
xM*(u[1]/xl1+u[2]/xpl2-u[3]/xm)
Fcn3
psim
Out_psim
Dpsi
Memory
Look-Up
Table
v2p-
Mux1
wb*(u[2] -(rp2/xpl2)*(u[1]-u[3]))
Fcn2
psi2'
psi2'_
Mux2
(u[1]-u[2])/xpl2
Fcn5
i2'
Out_i2'
Load Module
s1c/Load Module *
File Edit View Simulation
Format Tools Help
out_1
-RH
HGR
in_1
100%

نتائج المحاكاة :

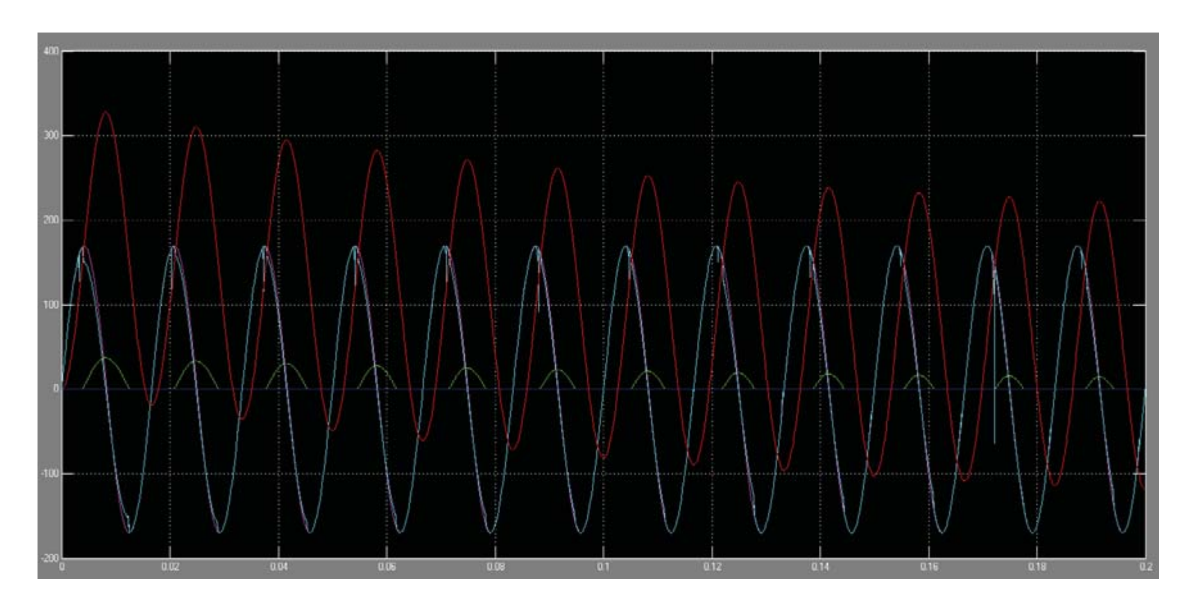
١. حالة القصر

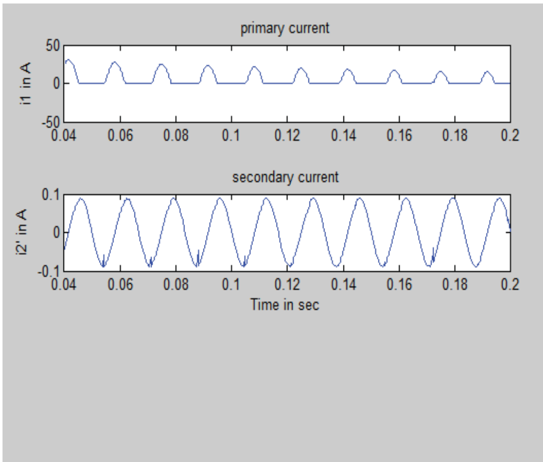


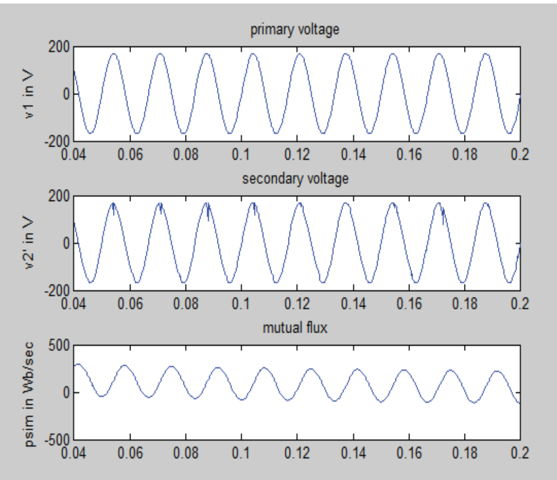




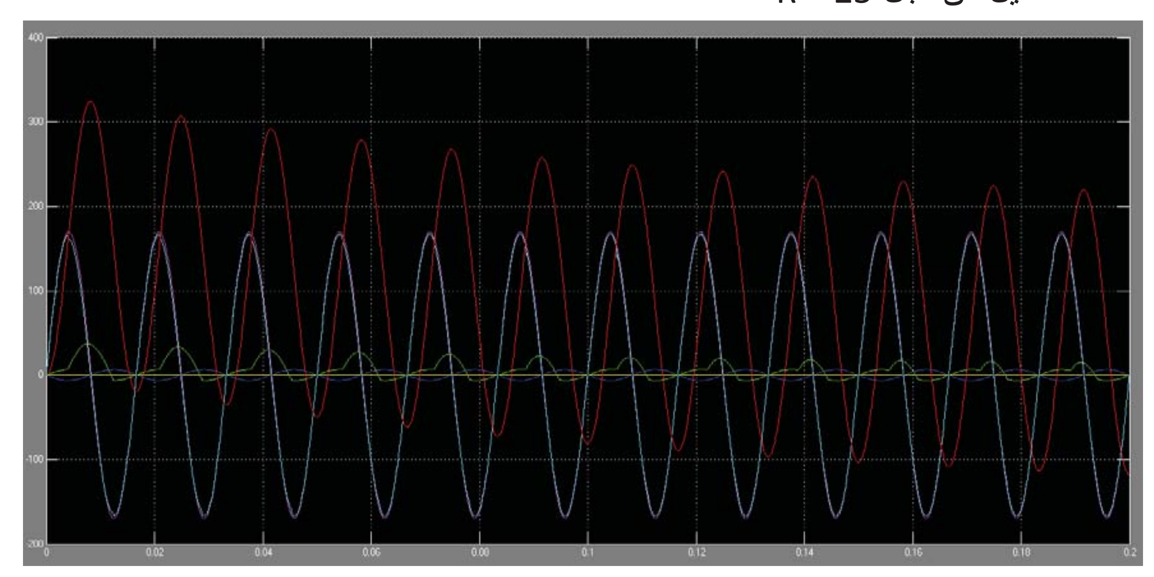
٢. حالة اللا حمل

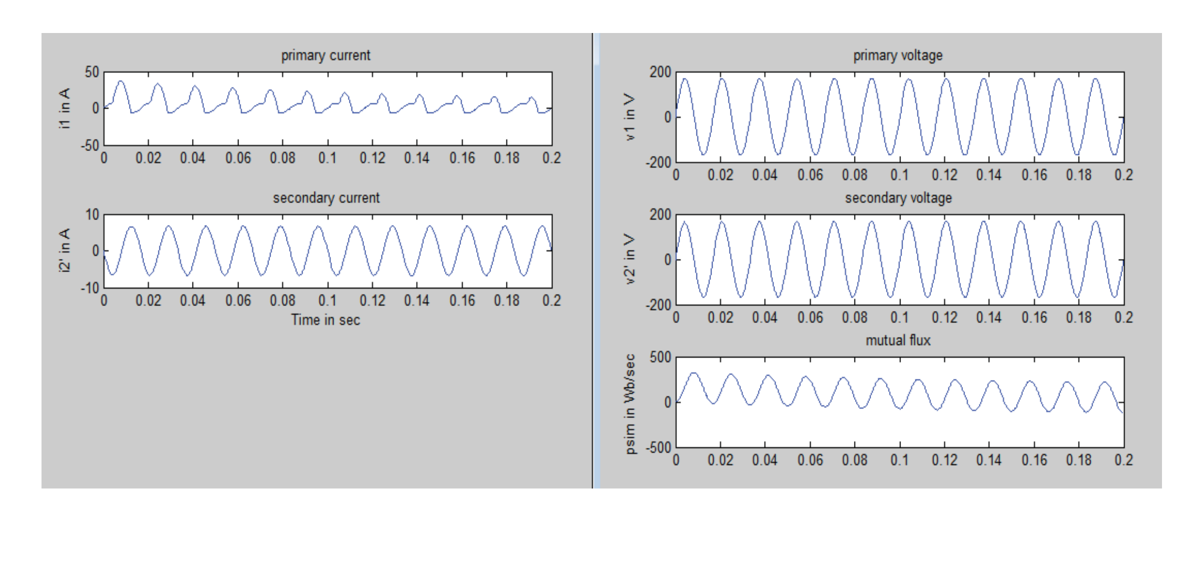






٣. حالة التحميل من أجل R = 25





٥,٢,١٤ نمذجة المحولة ثلاثية الطور

١,٥,٢,١٤ التوصيل نجمي – نجمي

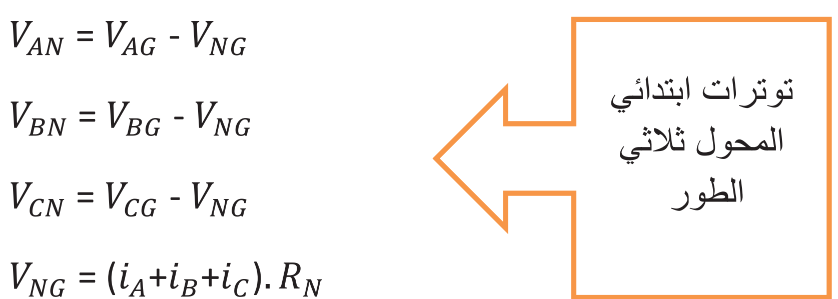
في هذه الحالة يتم تأريض نقطة النجمة في الملف الابتدائي بمقاومة أما نقطتا النجمة للثانوي والمنبع تؤرضان مباشرة إلى الأرض ، كما أن توتر نقطة الحيادي للابتدائي يكون عائماً من وجهة نظر تأريض النظام وذلك لأن التأريض غير مباشر.

$V_{AN} = V_{AG} - V_{NG}$

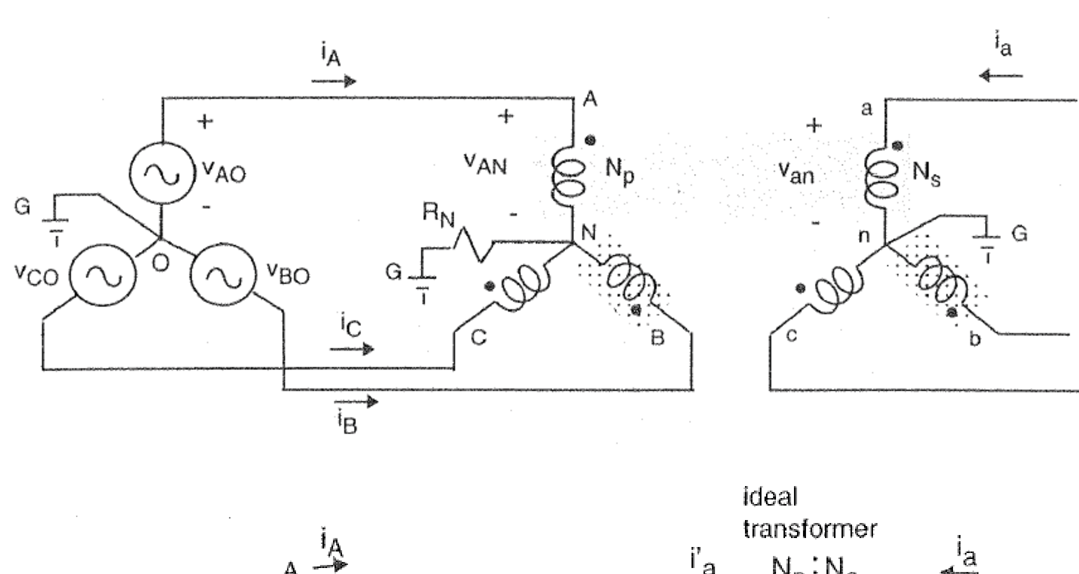
$V_{BN} = V_{BG} - V_{NG}$

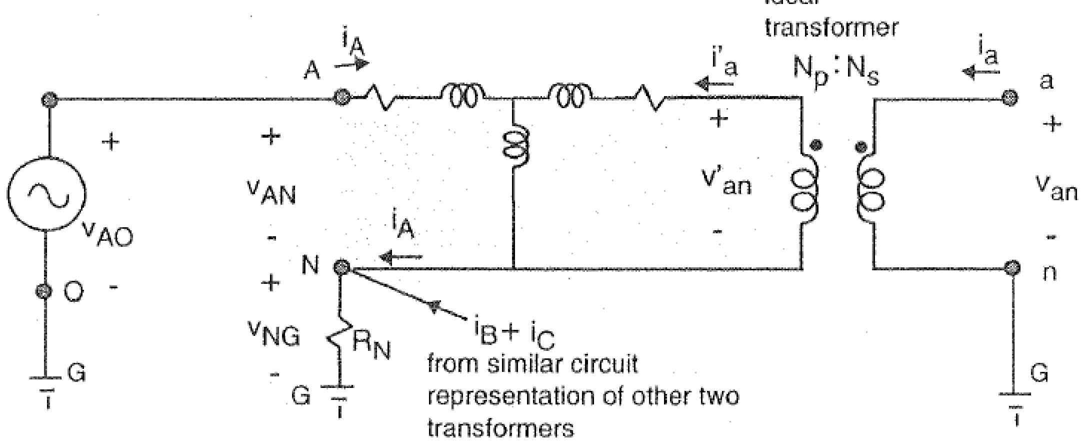
$V_{CN} = V_{CG} - V_{NG}$

$V_{NG} = (i_A+i_B+i_C).R_N$



الشكل التالي يبين مخطط التوصيل نجمي – نجمي لمحول ثلاثي الطور





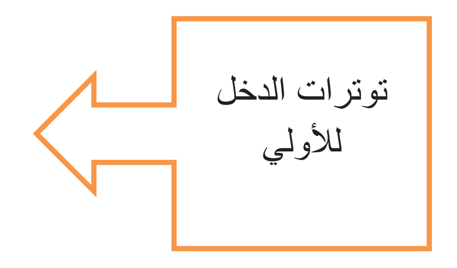
٢,٥,٢,١٤ التوصيل مثلثي – نجمي

في هذه الحالة يتم تأريض نقطة الحيادي في الثانوي بمقاومة ، وتكون نسبة ملفات الثانوي إلى الابتدائي مساوية لجذر ٣ مضروب بنسبة التحويل.

$V_{AB} = V_{Ao} - V_{Bo}$

$V_{BC} = V_{Bo} - V_{Co}$

$V_{CA} = V_{Co} - V_{Ao}$



$V_{an} = V_{an} - V_{nG}$

$V_{bn} = V_{bn} - V_{nG}$

$V_{cb} = V_{cn} - V_{nG}$

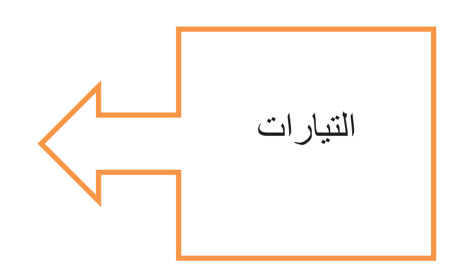
$V_{nG} = (i_a+i_b+i_c).R_n$



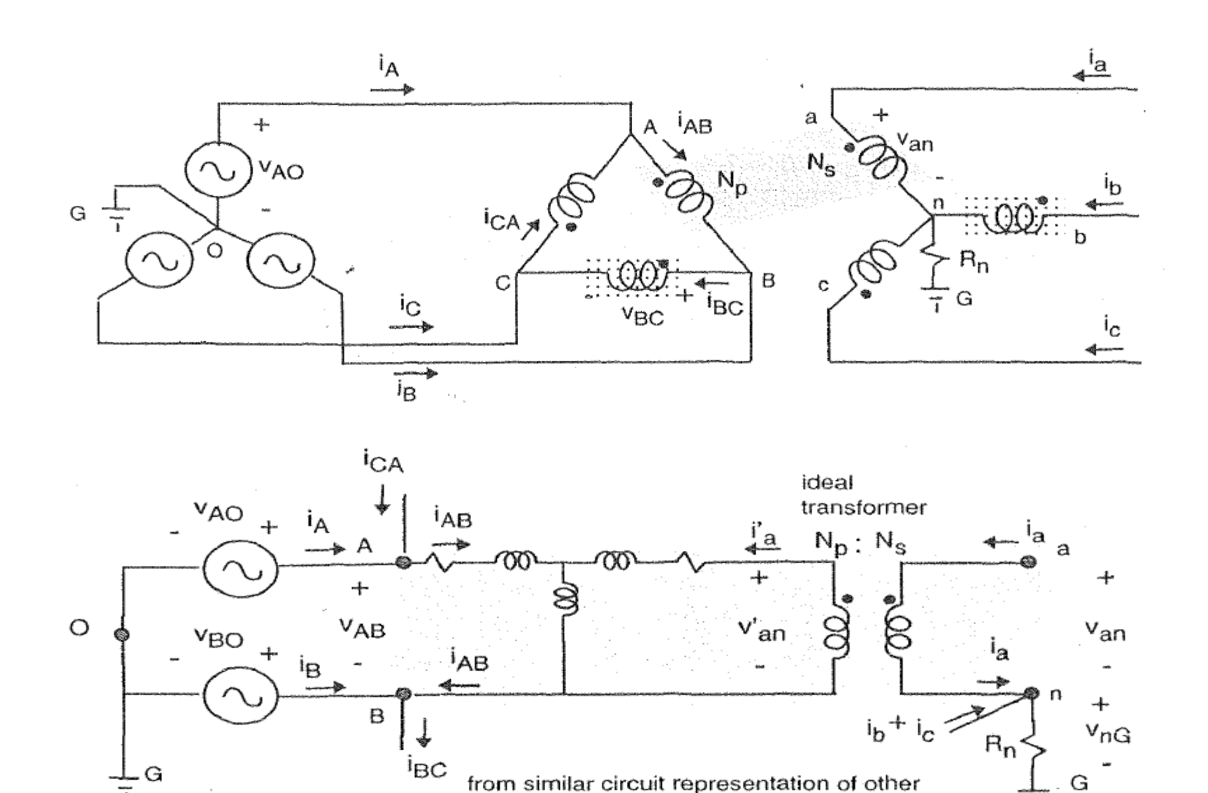
$i_A = i_{AB} - i_{CA}$

$i_B = i_{BC} - i_{AB}$

$i_C = i_{CA} - i_{BC}$



الشكل التالي يبين مخطط التوصيل مثلثي – نجمي لمحول ثلاثي الطور



والمخطط التالي يبين مخطط النمذجة لهذا المحول :

…M-File…

```
% محولة ثلاثية الطور
% جميع البارمترات من أجل محولة ذات ملفين

clear all;


Vrated = 120; % rms rated voltage
Srated = 1500; % rated VA
Frated = 60; % rated frequency in Hz
Zb = Vrated^2/Srated; % base impedance
wb = 2*pi*Frated; % base frequency
Vpk = Vrated*sqrt(2); % peak rated voltage
NpbyNs = 120/240; % nominal turns ratio
r1 = 0.25; % resistance of wdg 1 in ohms
rp2 = 0.134; % referred resistance of wdg 2 in ohms
xl1 = 0.056; % leakage reactance of wdg 1 in ohms
```

```
xpl2 = 0.056; % leakage reactance of wdg 1 in ohms
xm = 708.8; % unsaturated magnetizing reactance in ohms
xM = 1/(1/xm + 1/xl1 + 1/xpl2);

% mag. curve Dpsi versus psisat

Dpsi=[ -2454.6 -2412.6 -2370.5 -2328.5 -2286.4 -2244.4 -2202.3 ...
-2160.3 -2118.2 -2076.1 -2034.1 -1992.0 -1950.0 -1907.9 -1865.9 ...
-1823.8 -1781.8 -1739.7 -1697.7 -1655.6 -1613.6 -1571.5 -1529.5 ...
-1487.4 -1445.3 -1403.3 -1361.2 -1319.2 -1277.1 -1235.1 -1193.0 ...
-1151.0 -1108.9 -1066.9 -1024.8 -982.76 -940.71 -898.65 -856.60 ...
-814.55 -772.49 -730.44 -688.39 -646.43 -604.66 -562.89 -521.30 ...
-479.53 -438.14 -396.75 -355.35 -313.96 -272.56 -231.17 -192.60 ...
-154.04 -116.41 -81.619 -46.822 -19.566 0.0000 0.0000 0.0000
0.0000 ...
0.0000 0.0000 19.566 46.822 81.619 116.41 154.04 192.60 231.17 ...
272.56 313.96 355.35 396.75 438.14 479.53 521.30 562.89 604.66 ...
646.43 688.39 730.44 772.49 814.55 856.60 898.65 940.71 982.76 ...
1024.8 1066.9 1108.9 1151.0 1193.0 1235.1 1277.1 1319.2 1361.2 ...
1403.3 1445.3 1487.4 1529.5 1571.5 1613.6 1655.6 1697.7 1739.7 ...
1781.8 1823.8 1865.9 1907.9 1950.0 1992.0 2034.1 2076.1 2118.2 ...
2160.3 2202.3 2244.4 2286.4 2328.5 2370.5 2412.6 2454.6 ];

psisat=[ -170.21 -169.93 -169.65 -169.36 -169.08 -168.80 -168.52 ...
-168.23 -167.95 -167.67 -167.38 -167.10 -166.82 -166.54 -166.25 ...
-165.97 -165.69 -165.40 -165.12 -164.84 -164.56 -164.27 -163.99 ...
-163.71 -163.43 -163.14 -162.86 -162.58 -162.29 -162.01 -161.73 ...
-161.45 -161.16 -160.88 -160.60 -160.32 -160.03 -159.75 -159.47 ...
-159.18 -158.90 -158.62 -158.34 -157.96 -157.39 -156.83 -156.07 ...
-155.51 -154.57 -153.62 -152.68 -151.74 -150.80 -149.85 -146.08 ...
-142.31 -137.60 -130.06 -122.52 -107.44 -84.672 -42.336 0.0000 ...
0.0000 42.336 84.672 107.44 122.52 130.06 137.60 142.31 146.08 ...
149.85 150.80 151.74 152.68 153.62 154.57 155.51 156.07 156.83 ...
157.39 157.96 158.34 158.62 158.90 159.18 159.47 159.75 160.03 ...
160.32 160.60 160.88 161.16 161.45 161.73 162.01 162.29 162.58 ...
162.86 163.14 163.43 163.71 163.99 164.27 164.56 164.84 165.12 ...
165.40 165.69 165.97 166.25 166.54 166.82 167.10 167.38 167.67 ...
```

```
167.95 168.23 168.52 168.80 169.08 169.36 169.65 169.93 170.21 ];

% set up simulation parameters
tstop = 1.2; % stop time
Psi1o = 0; % initial value of wdg I flux linkage
Psip2o = 0; % initial value of wdg 2 flux  linkage
Rload = 120^2/1500 % referred impedance to primary side of
             % the 1.5 kVA, unity power factor load
                % connected across each secondary winding

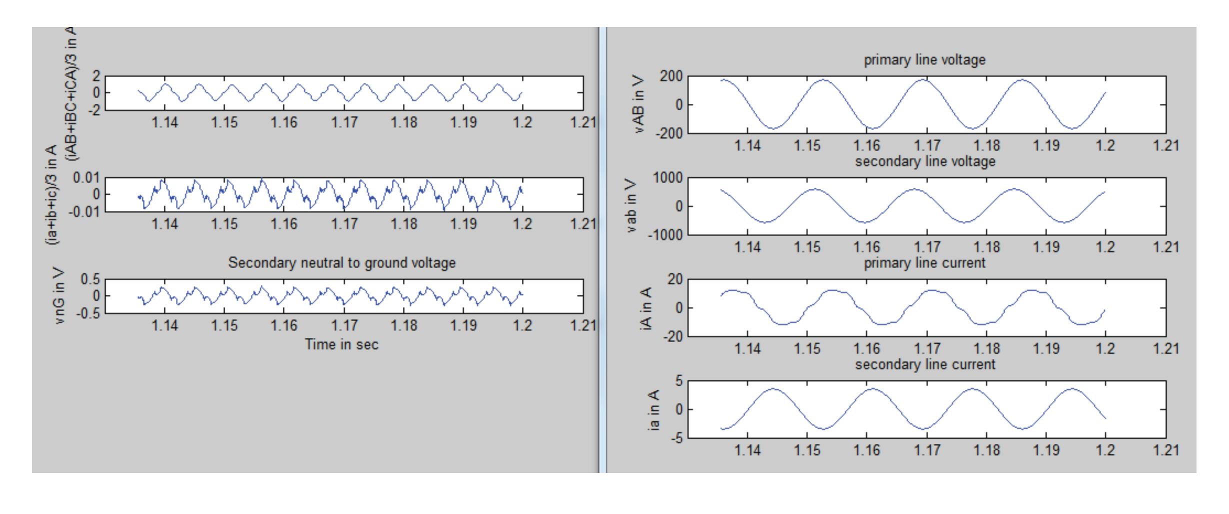
repeat_run = 'Y'; % set up repeat run flag
while repeat_run == 'Y'

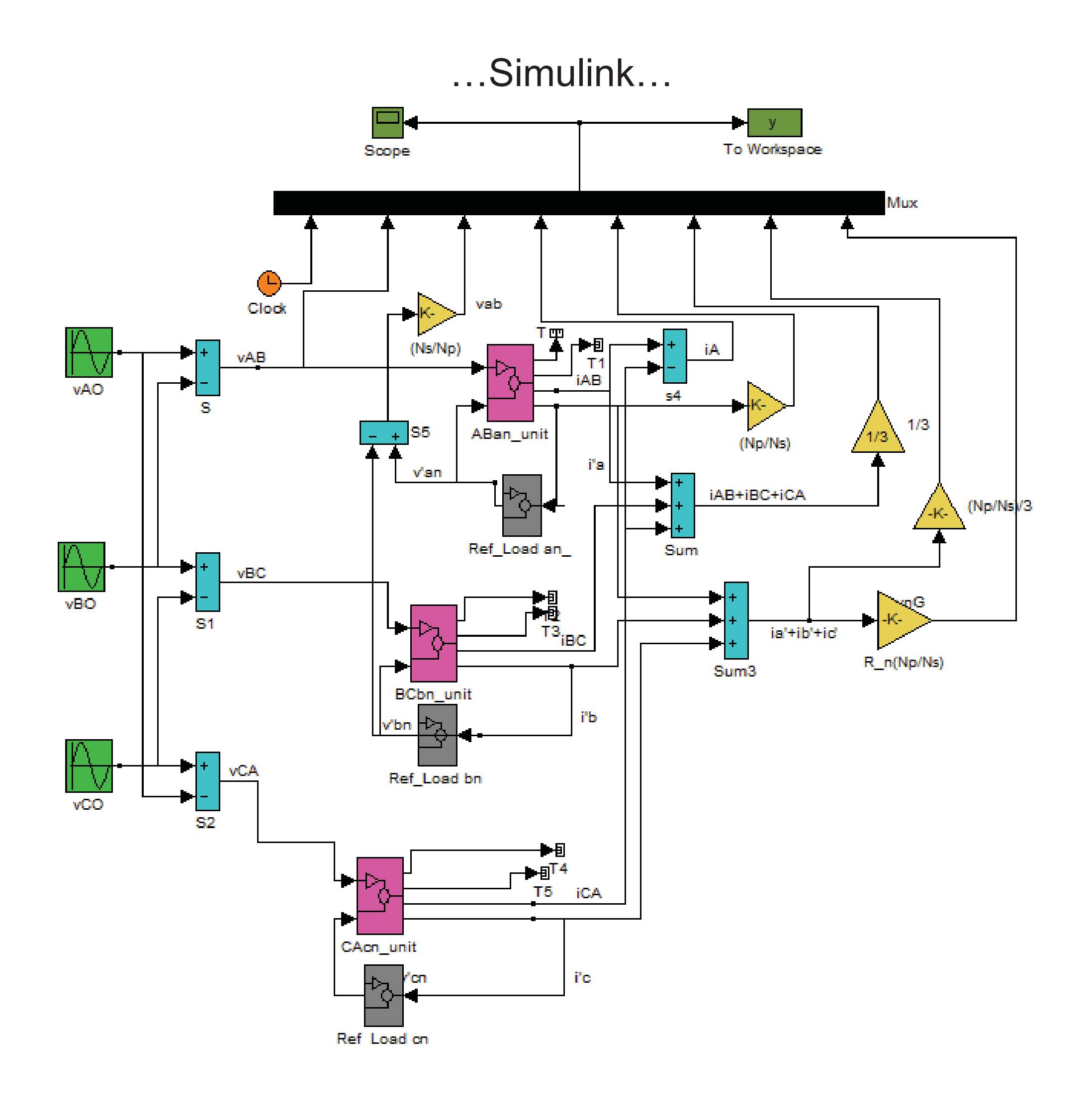
Rn = input('Enter ohmic value of neutral to ground resistor Rn > ');
disp('Run simulation and return for plots')
keyboard
clf;
subplot(4,1,1)
plot(y(:,1),y(:,2),'-')
ylabel('vAB in V')
title('primary line voltage')
subplot(4,1,2)
plot(y(:,1),y(:,3),'-')
ylabel('vab in V')
title('secondary line voltage')
subplot(4,1,3)
plot(y(:,1),y(:,4),'-')
ylabel('iA in A')
title('primary line current')
subplot(4,1,4)
plot(y(:,1),y(:,5),'-')
ylabel('ia in A')
title('secondary line current')
h2=figure;
subplot(4,1,1)
plot(y(:,1),y(:,6),'-')
ylabel('(iAB+iBC+iCA)/3 in A')
```

```
subplot(4,1,2)
plot(y(:,1),y(:,7),'-')
ylabel('(ia+ib+ic)/3 in A')
subplot(4,1,3)
plot(y(:,1),y(:,8),'-')
ylabel('vnG in V')
title('Secondary neutral to ground voltage')
xlabel('Time in sec')
disp('Save plots before typing return')
keyboard
close(h2)

% prompt for repeat with new system condition
repeat_run =  input('Repeat with new system condition? Y/N: ','s');
if isempty(repeat_run) % if empty return a No to terminate
repeat_run = 'N';
end
end % while repeat_run
```

نتائج المحاكاة :





٣,١٤ نمذجة ومحاكاة المحركات التحريضية

١,٣,١٤ جدول الرموز المستخدمة

r_s : مقاومة ملفات الثابت.

r_r : مقاومة ملفات الدائر.

r_f : مقاومة ملفات الحقل على المحور المباشر.

r_g : مقاومة ملفات الحقل على المحور العمودي.

r_{kd} : مقاومة ملفات الإخماد على المحور المباشر.

r_{kq} : مقاومة ملفات الإخماد على المحور العمودي.

L_{aa}, L_{bb}, L_{cc} : المفاعلة التحريضية الذاتية.

L_{ab}, L_{ca}, L_{bc} : المفاعلة التحريضية المتبادلة.

L_{ls} : المحارضة التسريبية لملفات الثابت.

L_{lf} : المحارضة التسريبية للملفات على المحور المباشر.

L_{lg} : المحارضة التسريبية للملفات على المحور العمودي.

L_{lkd} : المحارضة التسربية لملفات الإخماد على المحور المباشر.
L_{lkq} : المحارضة التسربية لملفات الإخماد على المحور العمودي.
L_{md} : المحارضة المغناطيسية للثابت على المحور المباشر.
L_{mq} : المحارضة المغناطيسية للثابت على المحور العمودي.
L_{mf} : المحارضة المغناطيسية لملفات الحقل على المحور المباشر.
L_{mg} : المحارضة المغناطيسية لملفات الحقل على المحور العمودي.
L_{mkd} : المحارضة المغناطيسية لملفات الإخماد على المحور المباشر.
L_{mkq} : المحارضة المغناطيسية لملفات الإخماد على المحور العمودي.
L_{ss} : المفاعلة التحريضية الثابت.
L_r : المفاعلة التحريضية الدائر.
L_{sr} : المفاعلة التحريضية المتبادلة بين الثابت والدائر.
F : القوة المحركة المغناطيسية.
λ : السيالة التسريبية لكل لفة.
λ_{md} : السيالة المغناطيسية.
φ : السيالة التسريبية الكلية.
φ_d, φ_q : مركبات السيالة على المحورين المتعامدين.
θ_r : الزاوية بين المحور العمودي (q) ومحور الطور الأول (a).
v_q, v_d : مركبات التوتر على المحورين المتعامدين.
v'_{kd}, v'_{kq} : مركبات توتر ملفات الإخماد على المحاور المتعامدة.

T_{em} : العزم الكهرطيسي (الكهرومغناطيسي).

سنناقش للمحرك التحريضي المواضيع التالية :

- مميزات المحرك التحريضي (خصائص التشغيل) {سرعة-تيار ، سرعة-عزم ، سرعة استطاعة و سرعة-مردود}.
- إقلاع وكبح المحركات التحريضية.
- سنناقش أيضاً بعض حالات المحرك التحريضي في حال وجود ممانعة بين نجمي ثابت المحرك التحريضي ونجمي الشبكة.
- المحرك التحريضي أحادي الطور.

١٤,٣,٢الانتقال من نظام المحاور (a,b,c) لنظام المحاور المتعامدة (d,q) {تحويل بارك}

إن الغاية من الانتقال بين المستويات هو :

١- التخلص من التداخل بين كافة بارامترات المحرك التحريضي.
٢- تقليل عدد المعادلات الواصفة للنظام وبالتالي تسهيل الحسابات.

ملاحظة : عملية النسب يمكن أن تكون لثابت المحرك أو للدائر.

- مصفوفة التحويل بين المحاور ....

$$[f_{dq0}] = [T_{dq0}(\theta)][f_{abc}]$$

$$[T_{dq0}(\theta)] = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

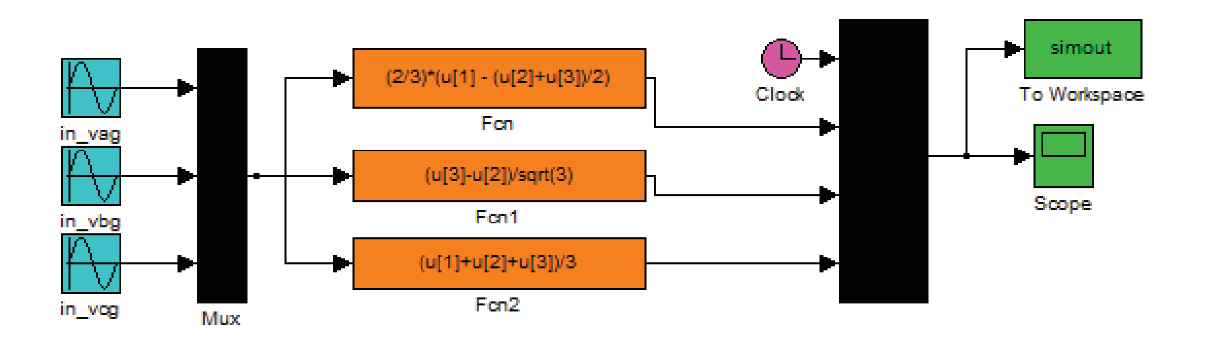
$$[T_{dq0}(\theta)]^{-1} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 1 \\ \cos(\theta - \frac{2\pi}{3}) & \sin(\theta - \frac{2\pi}{3}) & 1 \\ \cos(\theta + \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) & 1 \end{bmatrix}$$

لو جعلنا قيمة $(\theta = 0)$ فإن هذه الزاوية ستعطيني في كل لحظة قيمة الزاوية للسيالة الدوارة على المحور المباشر (d) المفروض وتصبح المصفوفة ....

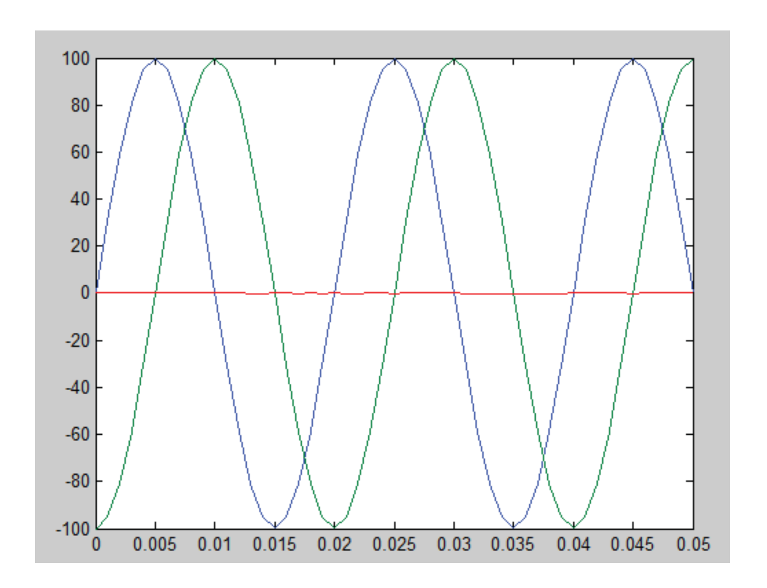
$$[T_{dq0}(0)] = \begin{bmatrix} \frac{2}{3} & \frac{-1}{3} & \frac{-1}{3} \\ 0 & \frac{-1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix}$$

$$[f_{dq0}] = [T_{dq0}(0)][f_{abc}]$$

نموذج الماتلاب الواصف لهذه المصفوفة :

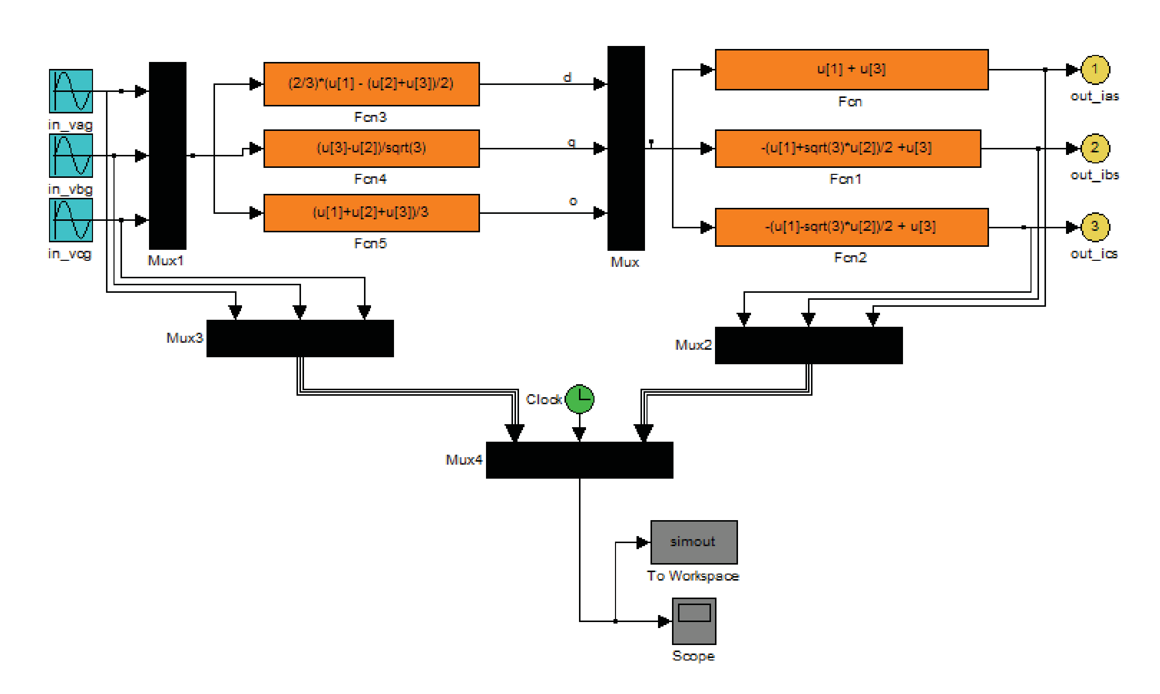


النتائج :



نلاحظ من النتائج أن المركبة الصفرية معدومة تقريباً (في حالات العمل النظامية – المستقرة) ولذلك يتم إهمال هذه المركبة في الحسابات.

كما أننا يمكننا العودة من نموذج المحاور (d,q) إلى النموذج ثلاثي الطور وذلك كما يبين النموذج التالي :

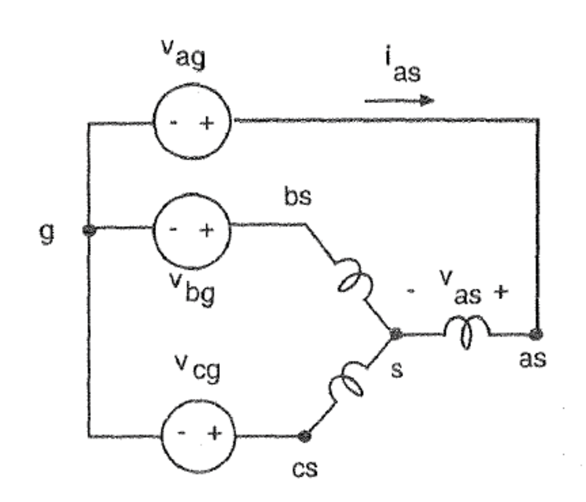


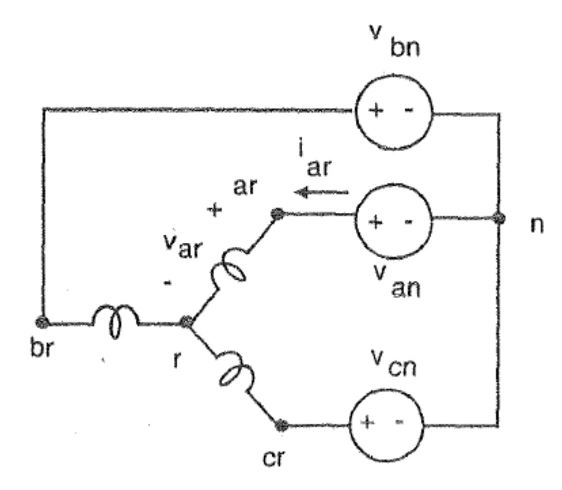
والمصفوفة الناتجة بالنتيجة تكون كما بالشكل :

$$\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ -0.5 & \frac{-\sqrt{3}}{2} & 1 \\ -0.5 & \frac{+\sqrt{3}}{2} & 1 \end{bmatrix} \begin{bmatrix} i_d \\ i_q \\ i_o \end{bmatrix}$$

٣,٣,١٤نمذجة ومحاكاة المحرك التحريضي

Simulation of An Induction motor in the Stationary Reference Frame





$$v_{as} = v_{ag} - v_{sg}$$
$$v_{bs} = v_{bg} - v_{sg}$$
$$v_{cs} = v_{cg} - v_{sg}$$

أو :

$$3v_{sg}=(v_{as}+v_{bs}+v_{cs}) - (v_{ag}+v_{bg}+v_{cg})$$

بإعادة تشكيل المعادلة نجد :

$$v_{sg}= R_{sg}(i_{as}+i_{bs}+i_{cs}) + L_{sg}\frac{d}{dx}(i_{as}+i_{bs}+i_{cs}) = 3(R_{sg}+L_{sg}\frac{d}{dx})i_{0s}$$

حيث :

R_{sg} ، L_{sg} : الحثية والمقاومة بين النقطتين (s,g).....

ولنقوم بتحويل المعادلة الاخيرة للمحورين المتعامدين (بالنسب للثابت) باعتبار :

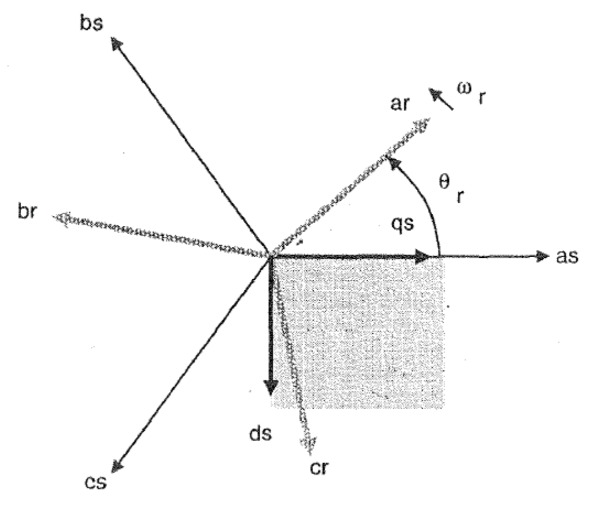
١. المحورين المتعامدين محازيين لمحور الطور الأول (a).
٢. $w = 0$.
٣. $\theta = 0$.

$$v_{qs}^{s} = \frac{2}{3}v_{as} - \frac{1}{3}v_{bs} - \frac{1}{3}v_{cs} = \frac{2}{3}v_{ag} - \frac{1}{3}v_{bg} - \frac{1}{3}v_{cg} - v_{sg}$$

$$v_{ds}^{s} = \frac{1}{\sqrt{3}}\left(v_{cs} - v_{bs}\right) = \frac{1}{\sqrt{3}}\left(v_{cg} - v_{bg}\right)$$

1

$$v_{0s} = \frac{1}{3}\left(v_{as}+v_{bs}+v_{cs}\right) = \frac{1}{3}\left(v_{ag}+v_{bg}+v_{cg}\right) - v_{sg}$$



عادة في نهاية النموذج يتم إعادة التحويل إلى المحاور الثلاثة (الانتقال من المحورين المتعامدين إلى نظام الثلاث محاور) وذلك كما في المعادلات التالية :

$$i_{as} = i_{qs}^{s} + i_{0s}$$

$$i_{bs} = \frac{-1}{2} i_{qs}^{s} - \frac{\sqrt{3}}{2} i_{ds}^{s} + i_{0s}$$

4

$$i_{cs} = \frac{-1}{2} i_{qs}^{s} + \frac{\sqrt{3}}{2} i_{ds}^{s} + i_{0s}$$

سنقوم الآن بالنسب للدائر :

$$v_{qr}^{\prime r} = \frac{2}{3} v_{ar}^{\prime} - \frac{1}{3} v_{br}^{\prime} - \frac{1}{3} v_{cr}^{\prime} = \frac{2}{3} v_{ar}^{\prime} - \frac{1}{3} v_{an}^{\prime} - \frac{1}{3} v_{cn}^{\prime} - v_{rn}^{\prime}$$

$$v_{dr}^{\prime r} = \frac{1}{\sqrt{3}}\left(v_{cr}^{\prime} - v_{br}^{\prime}\right) = \frac{1}{\sqrt{3}}\left(v_{cn}^{\prime} - v_{bn}^{\prime}\right)$$

2

$$v_{0r}^{\prime} = \frac{1}{3}\left(v_{ar}^{\prime}+v_{br}^{\prime}+v_{cr}^{\prime}\right) = \frac{1}{3}\left(v_{an}^{\prime}+v_{bn}^{\prime}+v_{cn}^{\prime}\right) - v_{rn}^{\prime}$$

سنقوم الآن بنسب معادلات الدائر للثابت :

$$v_{qr}^{\prime s} = v_{qr}^{\prime r}\cos\theta_{r}(t) + v_{dr}^{\prime r}\sin\theta_{r}(t)$$

3

$$v'^{s}_{dr} = -v'^{r}_{dr}\sin\theta_r(t) + v'^{r}_{dr}\sin\theta_r(t)$$

$$\theta_r(t) = \int_0^t w_r(t)dt + \theta_r(0)$$

وكذلك يمكننا العودة عكسياً لمجال الدائر (دون النسب للثابت) كما تبين المعادلات التالية :

$$i'^{r}_{qr} = i'^{s}_{qr}\cos\theta_r(t) - v'^{s}_{dr}\sin\theta_r(t)$$

$$v'^{r}_{dr} = i'^{s}_{qr}\sin\theta_r(t) + v'^{s}_{dr}\cos\theta_r(t)$$

5

والمعادلات التالية تبين كيف يمكن العودة من مجال الطورين المتعامدين إلى الأطوار الثلاثة للدائر :

$$i'_{ar} = i'^{r}_{qr} + i'_{0r}$$

$$i'_{br} = \frac{-1}{2} i'^{r}_{qr} - \frac{\sqrt{3}}{2} i'^{r}_{dr} + i'_{0r}$$

$$i'_{cr} = \frac{-1}{2} i'^{r}_{qr} + \frac{\sqrt{3}}{2} i'^{r}_{dr} + i'_{0r}$$

6

المعادلات الواصفة للمحرك في المحاور المتعامدة :

$$\psi^{s}_{qs} = w_b \int\{ v^{s}_{qs} + \frac{r_s}{x_{ls}}(\psi^{s}_{mq} - \psi^{s}_{qs})\}dt$$

$$\psi^{s}_{ds} = w_b \int\{ v^{s}_{ds} + \frac{r_s}{x_{ls}}(\psi^{s}_{md} - \psi^{s}_{ds})\}dt$$

$$i_{0s} = \frac{w_b}{x_{ls}}\int\{v_{0s} - i_{0s}r_s\}dt$$

7

$$\psi'^{s}_{qr} = w_b \int\{ v'^{s}_{qr} + \frac{w_r}{w_b}\psi'^{s}_{dr} + \frac{r'_r}{x'_{ls}}(\psi^{s}_{mq} - \psi'^{s}_{qr})\}dt$$

$$\psi'^{s}_{dr} = w_b \int\{ v'^{s}_{dr} - \frac{w_r}{w_b}\psi'^{s}_{qr} + \frac{r'_r}{x'_{ls}}(\psi^{s}_{md} - \psi'^{s}_{dr})\}dt$$

$$i'_{0r} = \frac{w_b}{x'_{lr}}\int\{v'_{0r} - i'_{0s}r'_r\}dt$$

8

$$\psi^{s}_{mq} = x_m(i^{s}_{qs} + i'^{s}_{qr})$$

$$\psi^{s}_{md} = x_m(i^{s}_{ds} + i'^{s}_{dr})$$

9

$$\psi_{qs}^{s} = x_{ls} i_{qs}^{s} + \psi_{mq}^{s} \qquad \psi_{qr}'^{s} = x_{ls}' i_{qr}'^{s} + \psi_{mq}^{s}$$

$$\psi_{ds}^{s} = x_{ls} i_{ds}^{s} + \psi_{md}^{s} \qquad \psi_{dr}'^{s} = x_{ls}' i_{dr}'^{s} + \psi_{md}^{s} \qquad (10)$$

$$i_{qs}^{s} = \frac{\psi_{qs}^{s} - \psi_{mq}^{s}}{x_{ls}}$$

$$i_{ds}^{s} = \frac{\psi_{ds}^{s} - \psi_{md}^{s}}{x_{ls}}$$

$$i_{qr}'^{s} = \frac{\psi_{qr}'^{s} - \psi_{mq}^{s}}{x_{lr}'}$$

$$i_{dr}'^{s} = \frac{\psi_{dr}'^{s} - \psi_{md}^{s}}{x_{lr}'}$$

$$\frac{1}{x_M} = \frac{1}{x_m} + \frac{1}{x_{ls}} + \frac{1}{x_{lr}'} \qquad (11)$$

$$\psi_{mq}^{s} = x_m\left(\frac{\psi_{qs}^{s}}{x_{ls}} + \frac{\psi_{qr}'^{s}}{x_{lr}'}\right) \qquad (12)$$

$$\psi_{md}^{s} = x_m\left(\frac{\psi_{ds}^{s}}{x_{ls}} + \frac{\psi_{dr}'^{s}}{x_{lr}'}\right)$$

معادلة العزم :

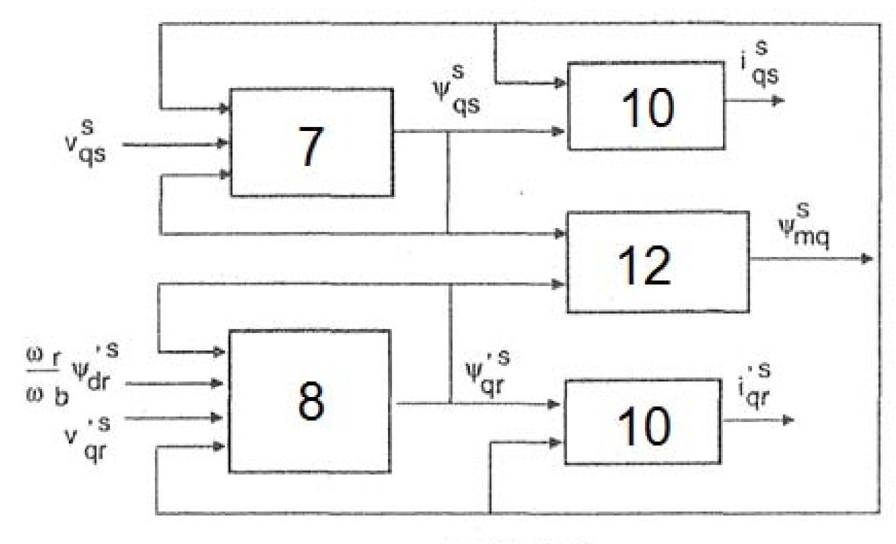
$$T_{em} = \frac{3}{2}\frac{P}{2w_b}(\psi_{ds}^{s}.\, i_{qs}^{s} - \psi_{qs}^{s}.\, i_{ds}^{s}) \qquad (13)$$

معادلة حركة الدائر :

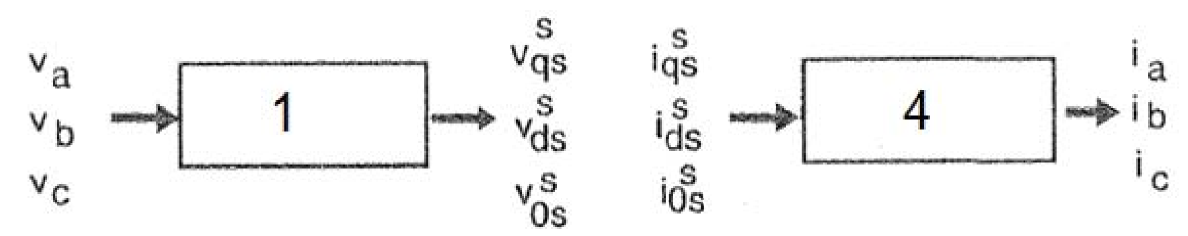
$$J\frac{dw_{rm}}{dt} = T_{em} + T_{mech} - T_{damp} \quad \text{N.m.} \qquad (14)$$

$$\frac{2J\, w_b}{P}\frac{d(\frac{w_r}{w_b})}{dt} = T_{em} + T_{mech} - T_{damp} \quad \text{N.m.} \qquad (15)$$

$$\text{H} = \frac{Jw_{bm}^{2}}{2S_b} \qquad (16)$$

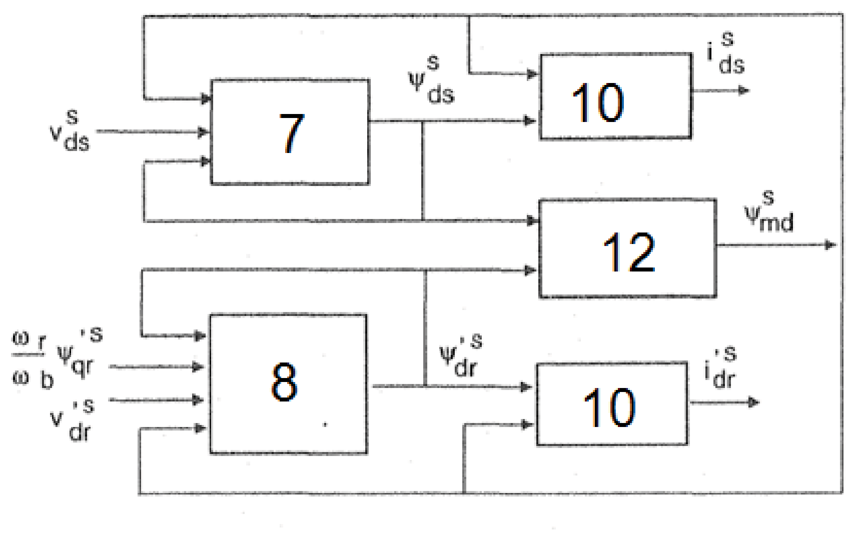


q-axis circuit

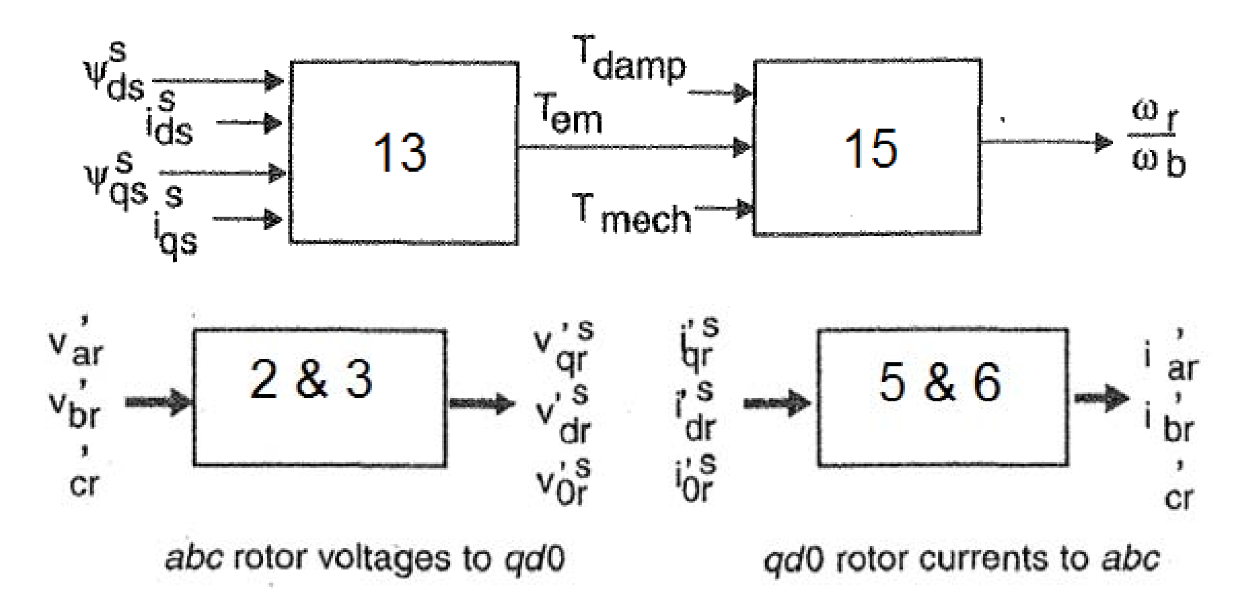


abc stator voltages to *qd*0

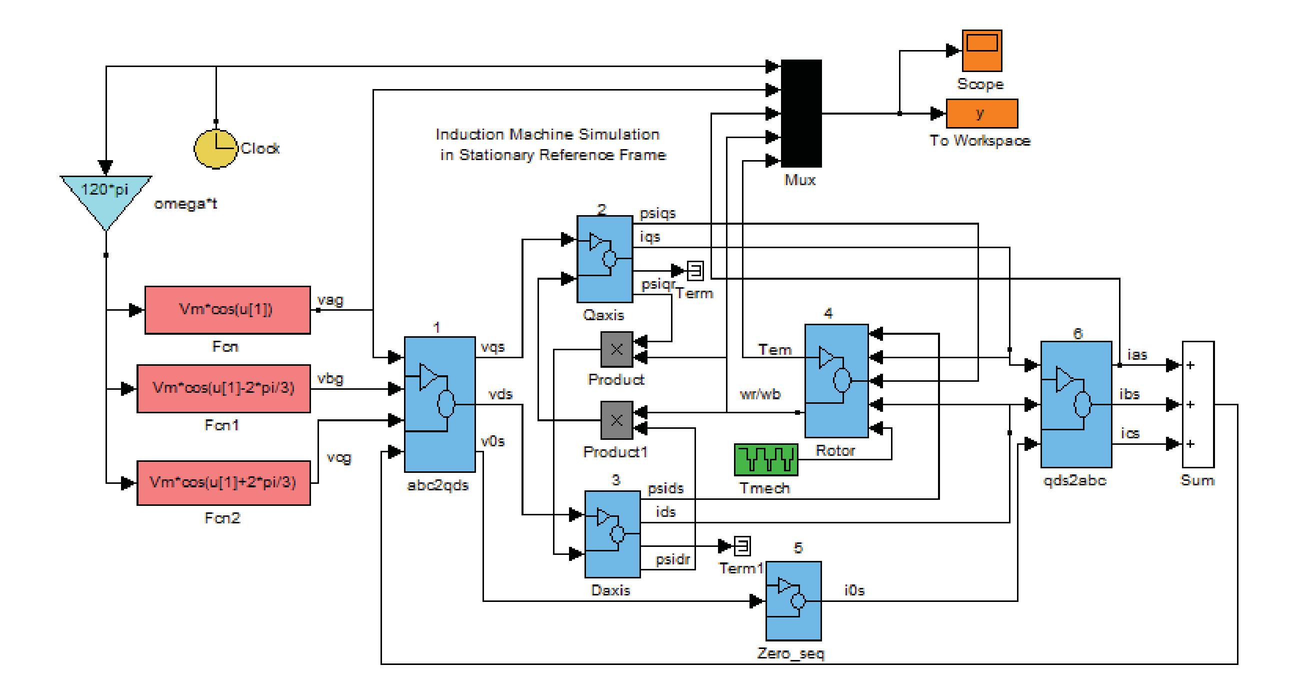
*qd*0 stator currents to *abc*



d-axis circuit



والأشكال التالية تبين نموذج المحاكاة باستخدام الماتلاب .

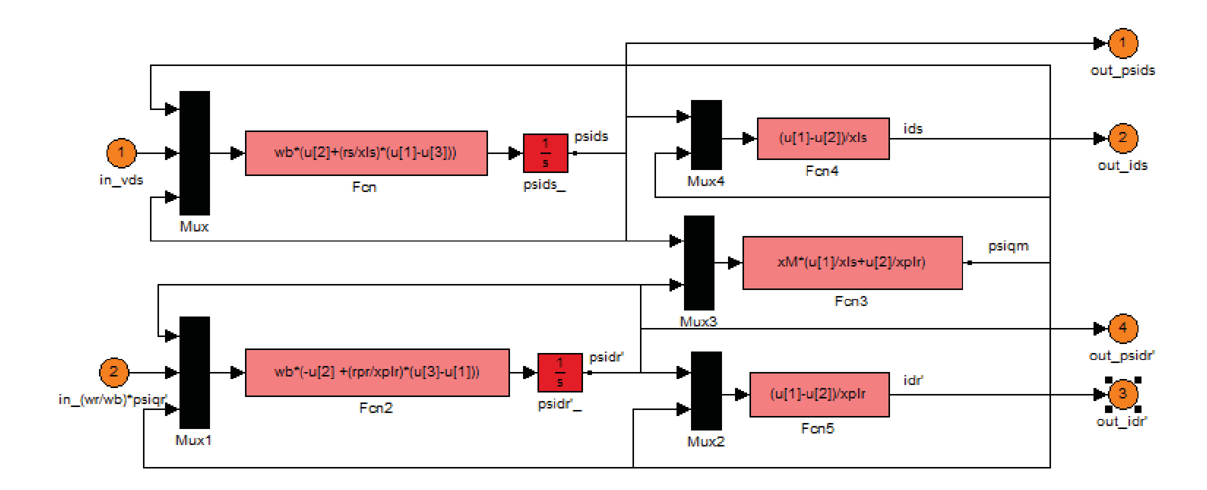


من الشكل نلاحظ أن نموذج محاكاة المحرك التحريضي يحوي العديد من النماذج الفرعية (Subsystem) وقد قم بترقيم كل منهم كما هو مبين ومحتوى كل منها مبين بالأشكال التالية ...

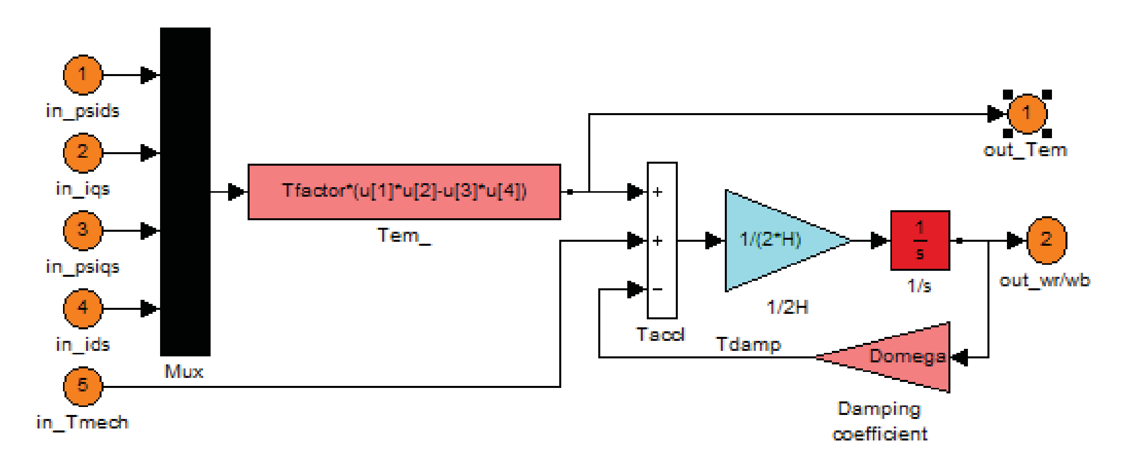
الصندوق (١)

الصندوق (٢)

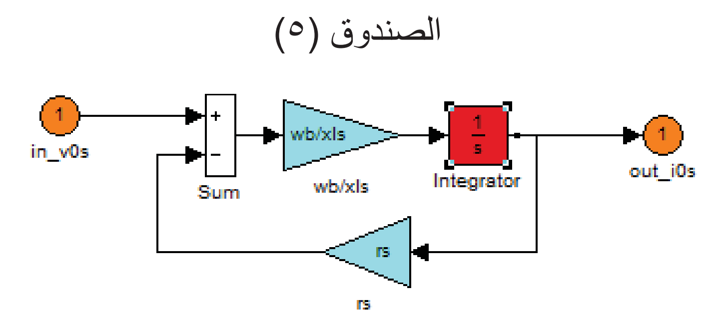
الصندوق (٣)



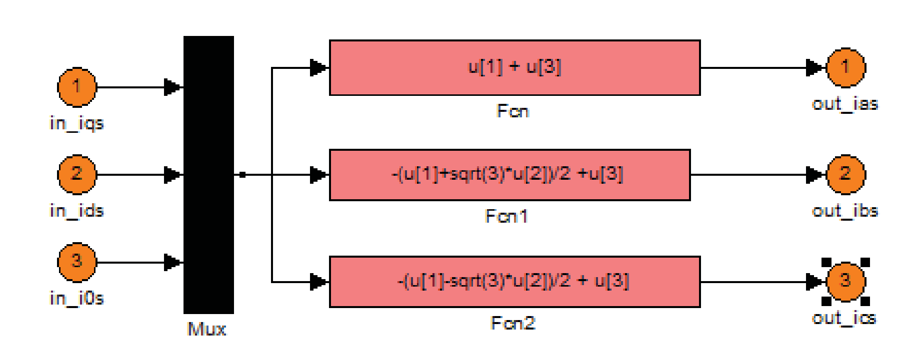
الصندوق (٤)



الصندوق (٥)



الصندوق (٦)



...Script File... P1

```
Sb = 750;   % rating in VA
Prated = 750;   % output power in W
Vrated = 200;   % rated line to line voltage in V
pf = 0.8;
Irated = Sb/(sqrt(3)*Vrated*pf); % rated rms current
P = 4;      % number of poles
frated = 60;    % rated frequency in Hz
wb = 2*pi*frated;   % base electrical frequency
we = wb;
wbm = 2*wb/P;   % base mechanical frequency
Tb = Sb/wbm;    % base torque
Zb = Vrated*Vrated/Sb;  %base impedance in ohms
Vm = Vrated*sqrt(2/3);  % magnitude of phase voltage
Vb = Vm; % base voltage
Tfactor = (3*P)/(4*wb); % factor for torque expression

rs = 3.35;  % stator resistance in ohms
xls = 6.94e-3*wb;% stator leakage reactance in ohms
xplr = xls;     % rotor leakage reactance
xm = 163.73e-3*wb;  %stator magnetizing reactance
rpr = 1.99; % referred rotor wdg resistance in ohms
xM = 1/(1/xm + 1/xls + 1/xplr);
J = 0.1;        % rotor inertia in kg m2
H = J*wbm*wbm/(2*Sb); % rotor inertia constant in secs.
Domega = 0; % rotor damping coefficient
```

...Script File... m

```
% Load three-phase induction motor parameters from this File ....
p1 % load  motor parameters from p1.m
% initialize to start from standstill with machine unexcited

Psiqso = 0; % stator q-axis total flux linkage
Psipqro = 0; % rotor q-axis total flux linkage
Psidso = 0; % stator d-axis total flux linkage
Psipdro = 0; % rotor d-axis total flux linkage
wrbywbo = 0; % pu rotor speed
tstop = 2; % use 2 sec simulation time for Fig. in text

% program time and output arrays of repeating sequence signal for
Tmech

tmech_time = [0 0.8 0.8 1.2 1.2 1.6 1.6 tstop];
tmech_value = [0 0 -0.5 -0.5 -1. -1. -0.5 -0.5]*Tb;

% Transfer to keyboard for simulation
disp('Set up for running s1.m or s3.m');
disp('Perform simulation then type return for plots');
keyboard
clf;
subplot(4,1,1)
plot(y(:,1),y(:,2),'-')
ylabel('vag in V')
title('stator phase to neutral voltage')
subplot(4,1,2)
plot(y(:,1),y(:,3),'-')
ylabel('ias in A')
axis([-inf inf -25 25]);
title('stator current')
subplot(4,1,3)
plot(y(:,1),y(:,5),'-')
ylabel('Tem in Nm')
```

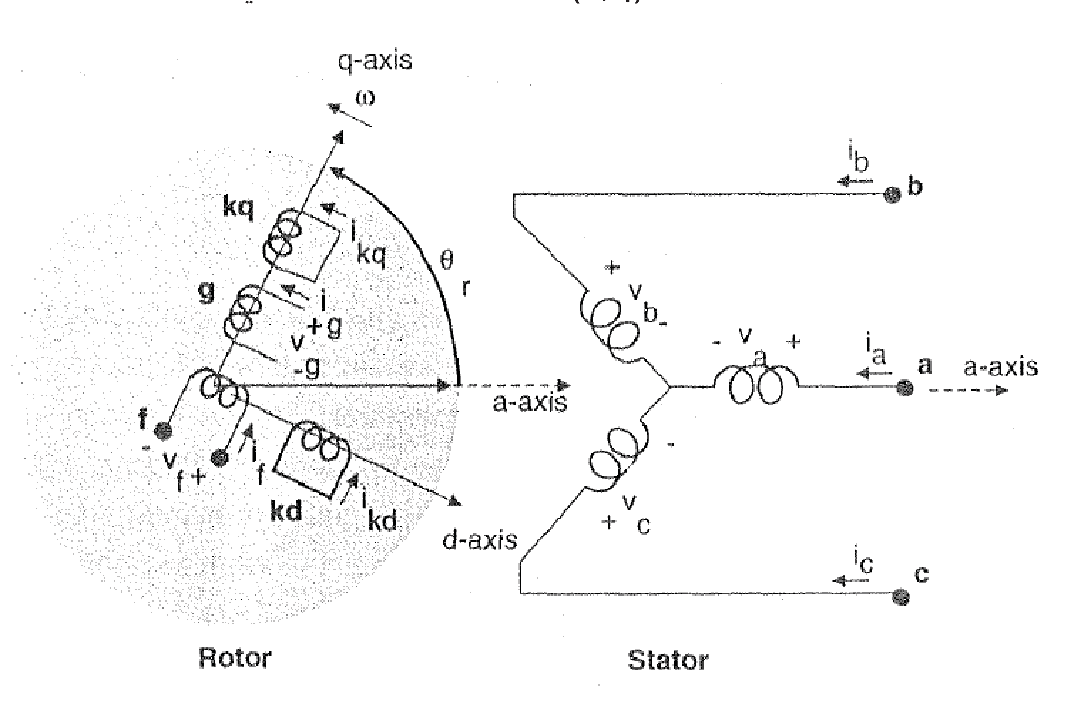
```
title('developed torque')
subplot(4,1,4)
plot(y(:,1),y(:,4),'-')
axis([-inf inf 0 1.2]);
ylabel('wr/wb')
xlabel('time in sec')
title('pu rotor speed')
```

٤,١٤ نمذجة ومحاكاة المحرك التزامنية

١,٤,١٤ مقدمة

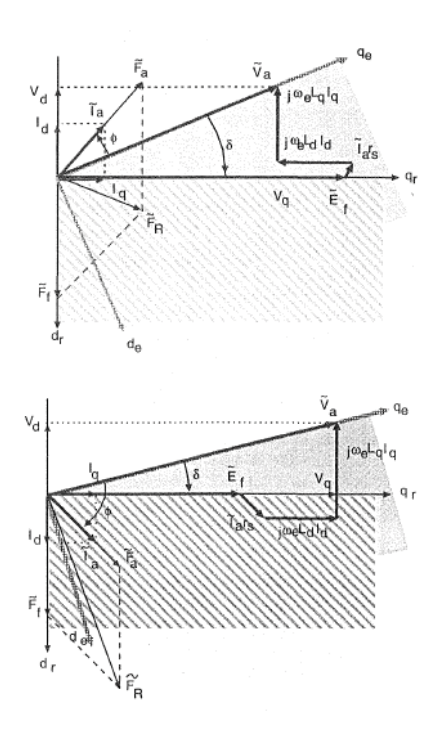
الآلات التزامنية هي تلك الآلات التي تدور بالسرعة التزامنية ، وعلى الرغم من أن تصميم الآلات التزامنية ثلاثية الطور ذو تكلفة باهظة بمقارنته مع الآلات التحريضية إلا أن أدائها العالي عند الاستطاعات الكبيرة يغطي على تكلفتها العالية ، حيث تستخدم الآلات التزامنية بشكل واسع في محطات التوليد الكبيرة وكذلك في نظم القيادة الضخمة.

وتمثل الآلة على المحورين المتعامدين (d,q) كما هو مبين بالشكل التالي :

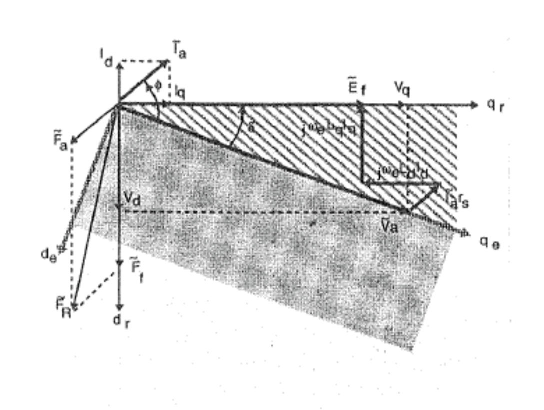


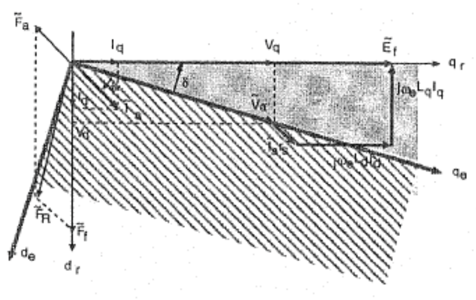
٢,٤,١٤ النموذج الرياضي للآلة التزامنية ثلاثية الطور

الأشكال التالية توضح الأنماط المختلفة لعمل الآلة التزامنية :



العمل كمحرك بمعامل استطاعة متقدم(الشكل العلوي) ومتأخر(الشكل السفلي)





العمل لمولد بمعامل استطاعة متقدم (الشكل العلوي) ومتأخر (الشكل السفلي)

الانتقال من نموذج المحاور الثلاثية (abc) إلى المحورين المتعامدين (dq) يتم وفق المعادلات التالية :

$$v_q^s = \frac{2}{3}v_a - \frac{1}{3}v_b - \frac{1}{3}v_c$$

$$v_d^s = \frac{1}{\sqrt{3}}(v_c - v_b) \qquad \boxed{1}$$

$$v_o = \frac{1}{3}(v_a + v_b + v_c)$$

$$v_q = v_q^s \cos\theta_r(t) - v_d^s \sin\theta_r(t) \qquad \boxed{٢}$$

$$v_d = v_q^s sin\theta_r(t) - v_d^s cos\theta_r(t)$$

حيث : $\theta_r(t) = \int_0^t w_r(t)\,dt + \theta_r(0)$ electrical radian $\boxed{3}$

أو يمكن إجراء التحويل بشكل آخر بخطوة واحدة وفق المعادلات التالية :

$\boxed{4}$

$$v_q = \frac{2}{3}\{v_a \cos\theta_r(t) + v_b \cos(\theta_r(t) - \frac{2\pi}{3}) + v_c \cos(\theta_r(t) + \frac{2\pi}{3})\}$$

$$v_d = \frac{2}{3}\{v_a sin\theta_r(t) + v_b sin(\theta_r(t) - \frac{2\pi}{3}) + v_c sin(\theta_r(t) + \frac{2\pi}{3})\}$$

$$v_o = \frac{1}{3}(v_a + v_b + v_c)$$

وللتعبير عن معادلات الجهد على المحاور المتعامدة (dq) كتكامل للسيالة التسربية للملفات ، فإن المعادلات السابقة والتي تصف جهد الثابت يمكن أن تحول لكي تحل معادلات السيالة التسربية للملفات. وفي حالة الآلة بملف حقل واحد فقط على المحور المباشر وزوج من ملفات الإخماد على المحورين المتعامدين ، فإن معادلات السيالة على المحاور المتعامدة تصبح :

$$\psi_d = w_b \int \{(v_q - \frac{w_r}{w_b}\psi_d + \frac{r_s}{x_{ls}}(\psi_{mq} - \psi_q)\}dt$$

$$\psi_q = w_b \int \{(v_d + \frac{w_r}{w_b}\psi_q + \frac{r_s}{x_{ls}}(\psi_{md} - \psi_d)\}dt$$

$$\psi_0 = w_b \int (v_0 - \frac{r_s}{x_{ls}}\psi_0)dt$$

$$\psi'_{kq} = \frac{w_b r'_{kq}}{x'_{lkq}} \int \left(\psi_{mq} - \psi'_{kq}\right) dt \qquad \boxed{5}$$

$$\psi'_{kd} = \frac{w_b r'_{kd}}{x'_{lkd}} \int (\psi_{md} - \psi'_{kd})\, dt$$

$$\psi'_f = \frac{w_b r'_f}{x'_{md}} \int \left(E_f + \frac{x_{md}}{x'_{lf}}(\psi_{md} - \psi'_f)\right) dt$$

حيث :

$$\psi_{mq} = w_b L_{mq}(i_q + i'_{kq})$$

$$\psi_{md} = w_b L_{md}(i_d + i'_{kd} + i'_f) \qquad \boxed{6}$$

$$E_f = x_{md}\frac{v'_f}{r'_f}$$

$$\psi_q = x_{ls} i_q + \psi_{mq}$$

$$\psi_d = x_{ls} i_d + \psi_{md}$$

$$\psi_0 = x_{ls} i_0$$

$$\psi'_f = x'_{lf} i'_f + \psi_{md}$$

$$\psi'_{kd} = x'_{lkd} i'_{kd} + \psi_{md}$$

$$\psi'_{kq} = x'_{lkq} i'_{kq} + \psi_{mq}$$

7

لابد من ملاحظة أن المعادلات العلوية هي من أجل المحرك ، والتي تكون التيارات فيها مع القطبية الموجبة للجهد المطبق على ملفات الثابت. وسنعبر عن السيالة التسربية المتشابكة كتابع للسيالة الكلية التسربية في الملفات بحيث تكون المعادلات :

$$\psi_{mq} = x_{MQ}\left(\frac{\psi_q}{x_{ls}} + \frac{\psi'_{kq}}{x'_{lkq}}\right)$$

$$\psi_{md} = x_{MD}\left(\frac{\psi_d}{x_{ls}} + \frac{\psi'_{kd}}{x'_{lkd}} + \frac{\psi'_f}{x'_{lf}}\right)$$

8

حيث :

$$\frac{1}{x_{MQ}} = \frac{1}{x_{mq}} + \frac{1}{x'_{lkq}} + \frac{1}{x_{ls}}$$

$$\frac{1}{x_{MD}} = \frac{1}{x_{md}} + \frac{1}{x'_{lkd}} + \frac{1}{x'_{lf}} + \frac{1}{x_{ls}}$$

9

وبعد الحصول على قيم السيالة التسربية للملفات والسيالة التشابكية التسربية على المحورين المتعامدين يمكن تحديد تيارات الملفات حسب المعادلات التالية :

$$i_q = \frac{\psi_q - \psi_{mq}}{x_{ls}}$$

$$i_d = \frac{\psi_d - \psi_{md}}{x_{ls}}$$

10

$$i'_{kd} = \frac{\psi'_{kd} - \psi_{md}}{x'_{lkd}}$$

$$i'_{kq} = \frac{\psi'_{kq} - \psi_{mq}}{x'_{lkq}}$$

$$i'_{f} = \frac{\psi'_{f} - \psi_{md}}{x'_{lf}}$$

التيارات على المحاور المتعامدة يمكن الحصول على القيم المكافئة لها على نموذج المحاور الثلاثية وفق المعادلات التالية :

$$i_q^s = i_q cos\theta_r(t) + i_d sin\theta_r(t)$$

$$i_d^s = -i_d sin\theta_r(t) + i_d cos\theta_r(t)$$

11

$$i_a = i_q^s + i_0$$

$$i_b = -\frac{1}{2} i_q^s - \frac{1}{\sqrt{3}} i_d^s + i_0$$

$$i_c = -\frac{1}{2} i_q^s + \frac{1}{\sqrt{3}} i_d^s + i_0$$

12

العزم الكهروميكانيكي الناتج عن ألة تحوي عدد من الأقطاب محدد في حال العمل كمحرك تعطي بالعلاقة :

$$T_{em} = \frac{P_{em}}{w_{rm}} = \frac{3}{2}\frac{P}{2}(\lambda_d i_q - \lambda_q i_d) = \frac{3}{2}\frac{P}{2w_b}(\psi_d i_q - \psi_q i_d)$$

13

١,٢,٤,١٤ المعادلات الواصفة لحركة الدائر :

في حالة العمل كمحرك ، العزم الدقيق الناتج عن التسارع يكون في اتجاه حركة الدائر (أي في اتجاه الدوران). في حال عمل الألة كمحرك فإن العزم الكهروميكانيكي الناتج يكون موجب بينما في حالة العمل كمولد فإن العزم الكهروميكانيكي الناتج يكون سالب، أما العزم الميكانيكي الخارجي فيكون سالب في حالة المحرك وموجب في حالة المولد حيث عندها يقاد المولد بوسيلة خارجية. أما عزم الاحتكاك فيكون معاكساً لحركة الدوران ، والمعادلة التالية تمثل العزم المتسارع كتابع لعزم العطالة

14

$$T_{em} + T_{mech} - T_{damp} = J\frac{dw_{rm}(t)}{dt} = \frac{2J}{P}\frac{dw_r(t)}{dt} \quad N.m$$

زاوية الدائر تعرف كما يلي :

$$\delta(t) = \theta_r(t) - \theta_e(t)$$

$$= \int_0^t \{w_r(t) - w_e\}dt + \theta_r(0) - \theta_e(0) \qquad 15$$

$$w_e = constant$$

$$\frac{d\{w_r(t) - w_e\}}{dt} = \frac{dw_r(t)}{dt} \qquad 16$$

$$w_r(t) - w_e = \frac{P}{2J}\int_0^t \left(T_{em} + T_{mech} - T_{damp}\right)dt \qquad 17$$

θ_r: زاوية الدائر بالنسبة للمحور العمودي أي في المجال المرجعي للدائر

θ_e: زاوية السيالة بالنسبة للمحور العمودي أي في المجال المرجعي للسيالة التزامنية

٢,٢,٤,١٤ التعبير بالقيم الواحدية لمعادلة العزم ومعادلة حركة الدائر :

عند دراسة نظم القدرة حيث يكون هناك العديد من المحولات وكذلك تجهيزات متعددة في الشبكة فلابد من أخذ نقطة أساس للعمل والعمل بالقيم الواحدية عندها (النسب للأساس).
دراستنا تتضمن فقط ألة تزامنية واحدة ولذلك لن نستفيد من جميع نواحي استخدام النسب.

ممانعة الأساس $\quad Z_b = \frac{V_b}{I_b} \quad \Omega$

عزم الأساس $\quad T_b = \frac{S_b}{w_{bm}} \quad N.m$

$$w_{bm} = \frac{2w_b}{P}$$

ومن المعادلة ١٣ يعبر عنها بالقيم الواحدية :

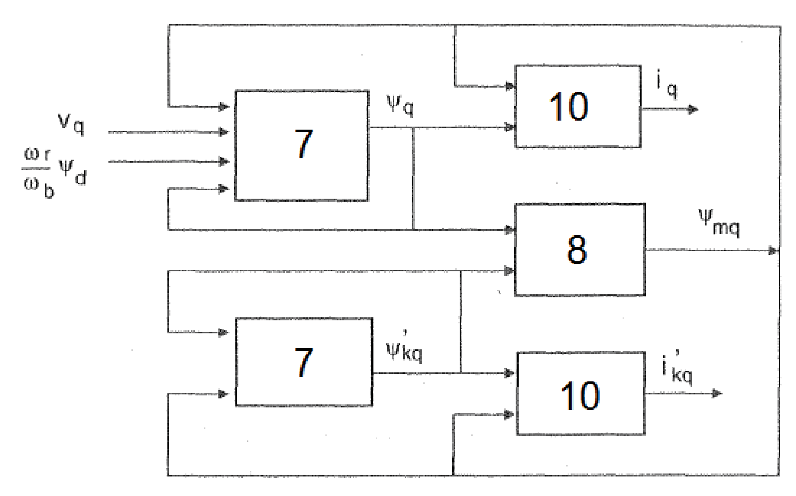
$$T_{em}(pu) = \frac{T_{em}}{T_b} = \frac{\frac{3}{2}\frac{P}{2w_b}(\psi_d i_q - \psi_q i_d)}{\frac{3}{2}\left(\frac{V_b I_b}{\frac{2}{P}w_b}\right)} \quad pu \qquad 17$$

$$T_{em}(pu) = \psi_{d(pu)} i_{q(pu)} - \psi_{q(pu)} i_{d(pu)}$$

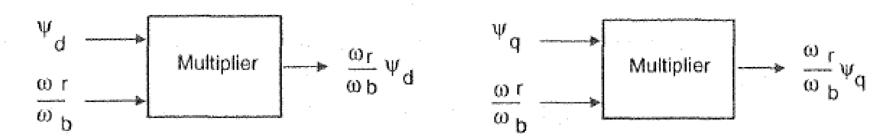
ومن المعادلة ١٤ يعبر عنها بالقيم الواحدية :

$$T_{em(pu)} + T_{mech(pu)} - T_{damp(pu)} = \left(\frac{1}{T_b}\right)\left(\frac{2J}{P}\right)\frac{dw_r}{dt} \quad pu \qquad (18)$$

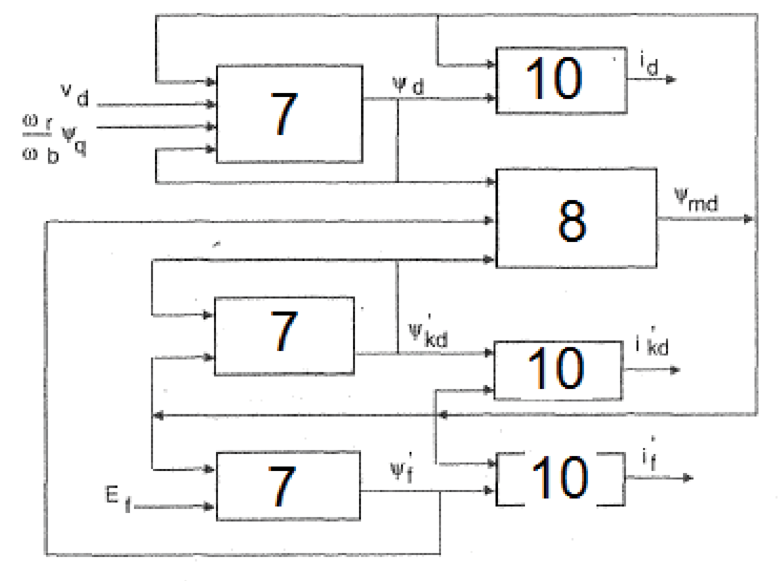
$$T_{em(pu)} + T_{mech(pu)} - T_{damp(pu)} = 2H\frac{d(^{w_r}/_{w_b})}{dt} = 2H\frac{d(^{w_r - w_e}/_{w_b})}{dt}$$



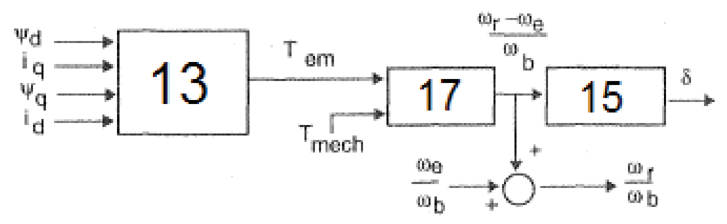
(a) q-axis circuit



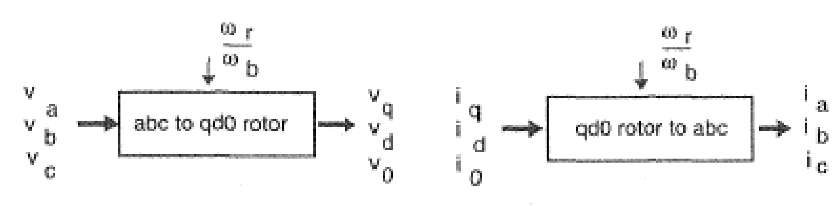
(b) Speed voltages



(c) *d*-axis circuit

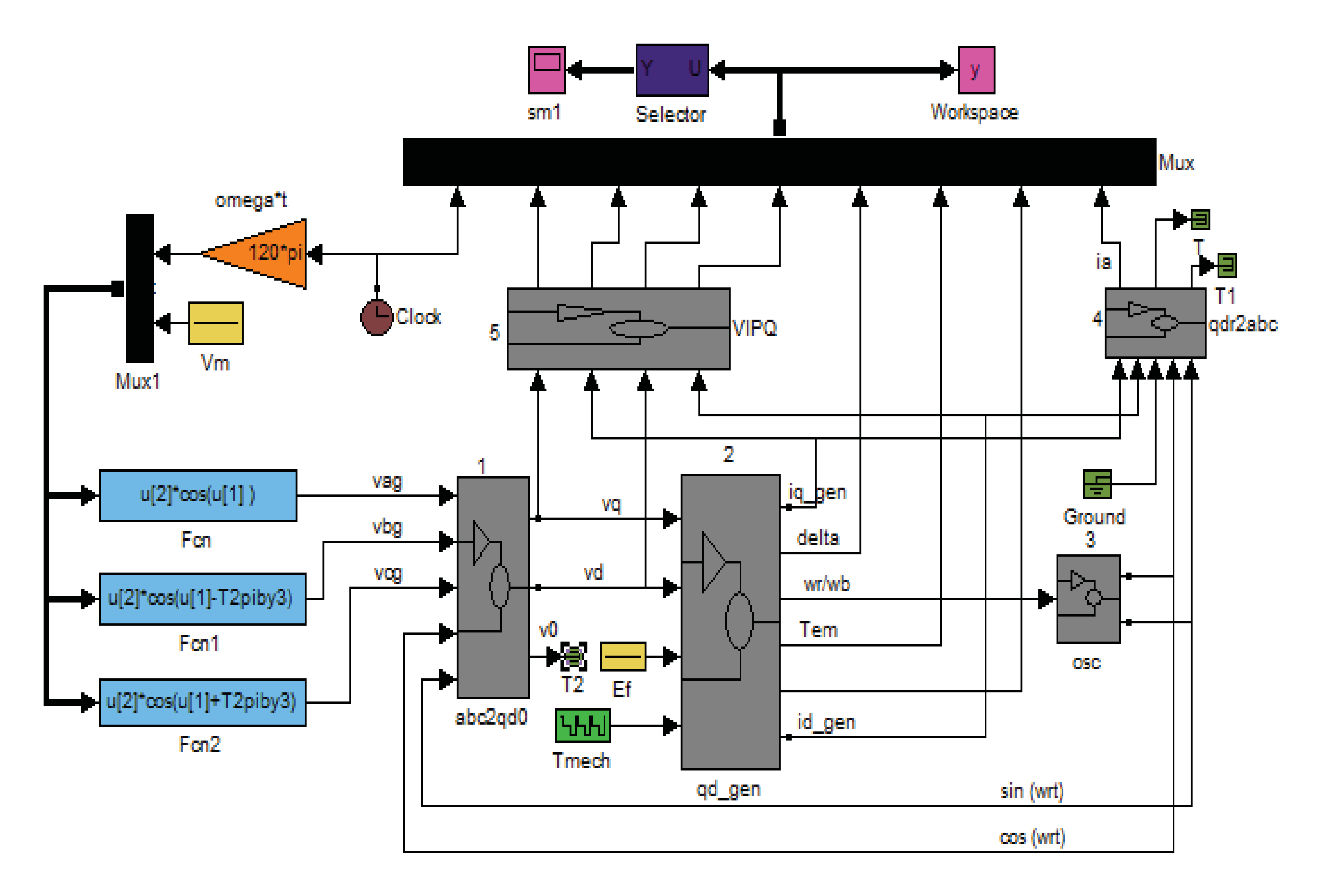


(d) Developed torque, speed, and angle

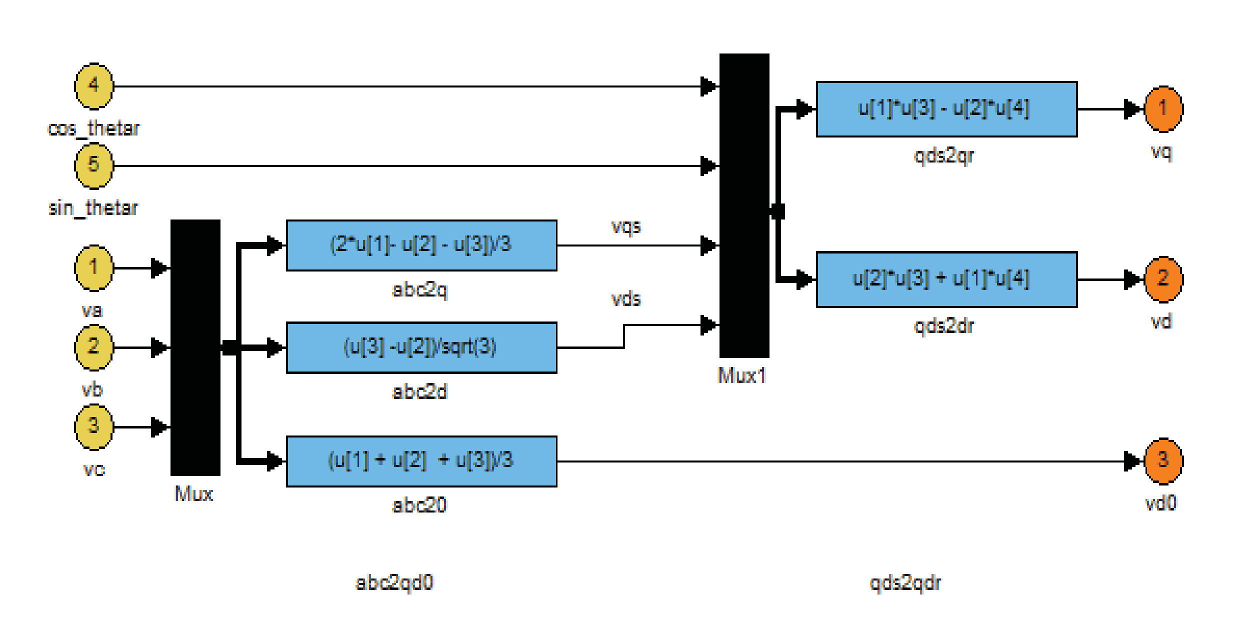


(e) Transform *abc* supply voltages to *qd*0

(f) Transform *qd*0 currents to *abc*


sm1
Selector
Workspace
y
Mux
omega*t
120*pi
Clock
Mux1
Vm
5
VIPQ
ia
T
T1
4
qdr2abc
1
2
u[2]*cos(u[1])
Fcn
u[2]*cos(u[1]-T2piby3)
Fcn1
u[2]*cos(u[1]+T2piby3)
Fcn2
vag
vbg
vcg
vq
vd
v0
T2
Ef
abc2qd0
Tmech
iq_gen
delta
wr/wb
Tem
id_gen
qd_gen
Ground
3
osc
sin (wrt)
cos (wrt)

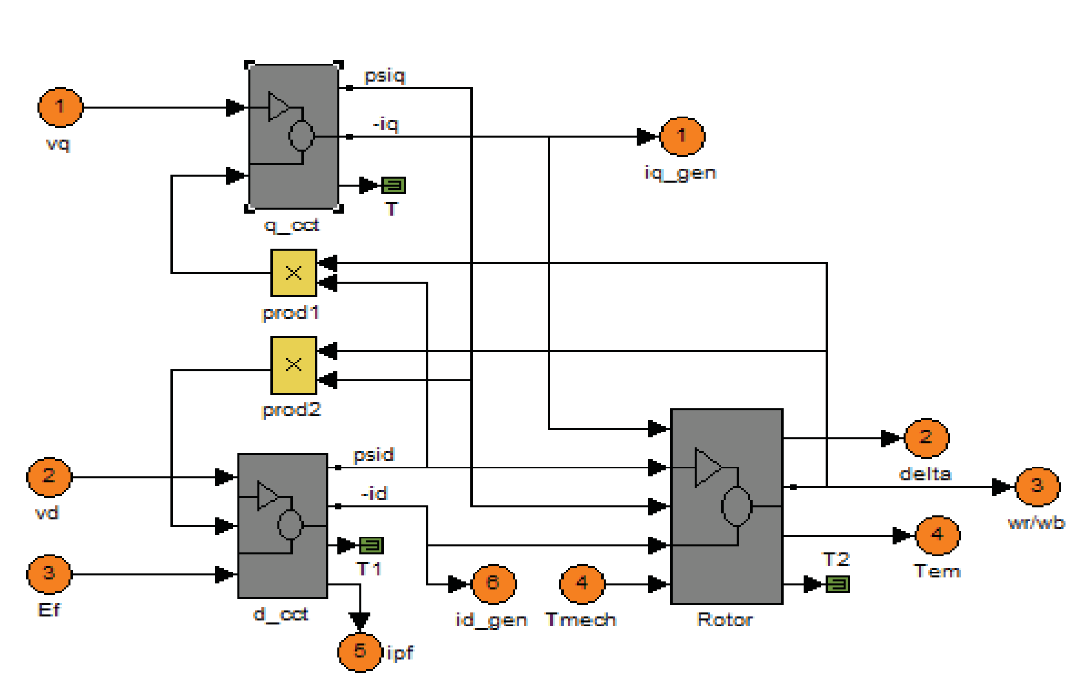
...١...



4
cos_thetar
5
sin_thetar
1
va
2
vb
3
vc
Mux
(2*u[1]- u[2] - u[3])/3
abc2q
(u[3] -u[2])/sqrt(3)
abc2d
(u[1] + u[2] + u[3])/3
abc20
vqs
vds
Mux1
u[1]*u[3] - u[2]*u[4]
qds2qr
u[2]*u[3] + u[1]*u[4]
qds2dr
1
vq
2
vd
3
vd0
abc2qd0
qds2qdr

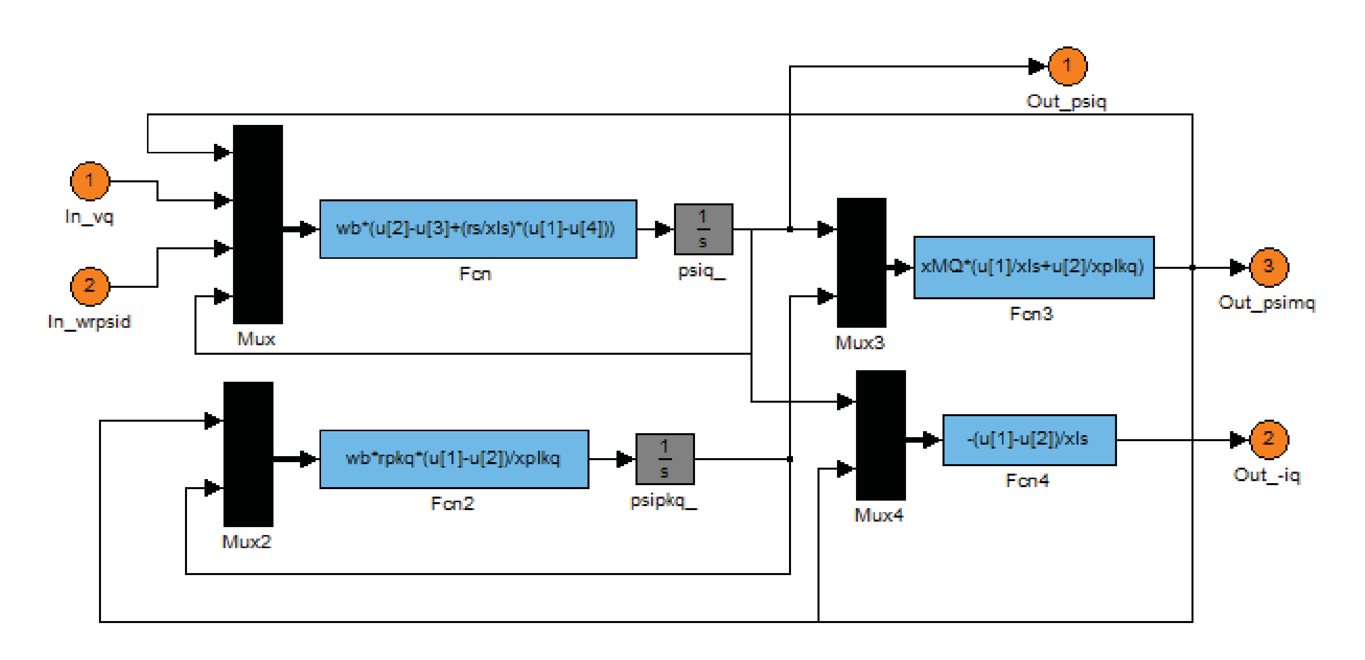


...٢...

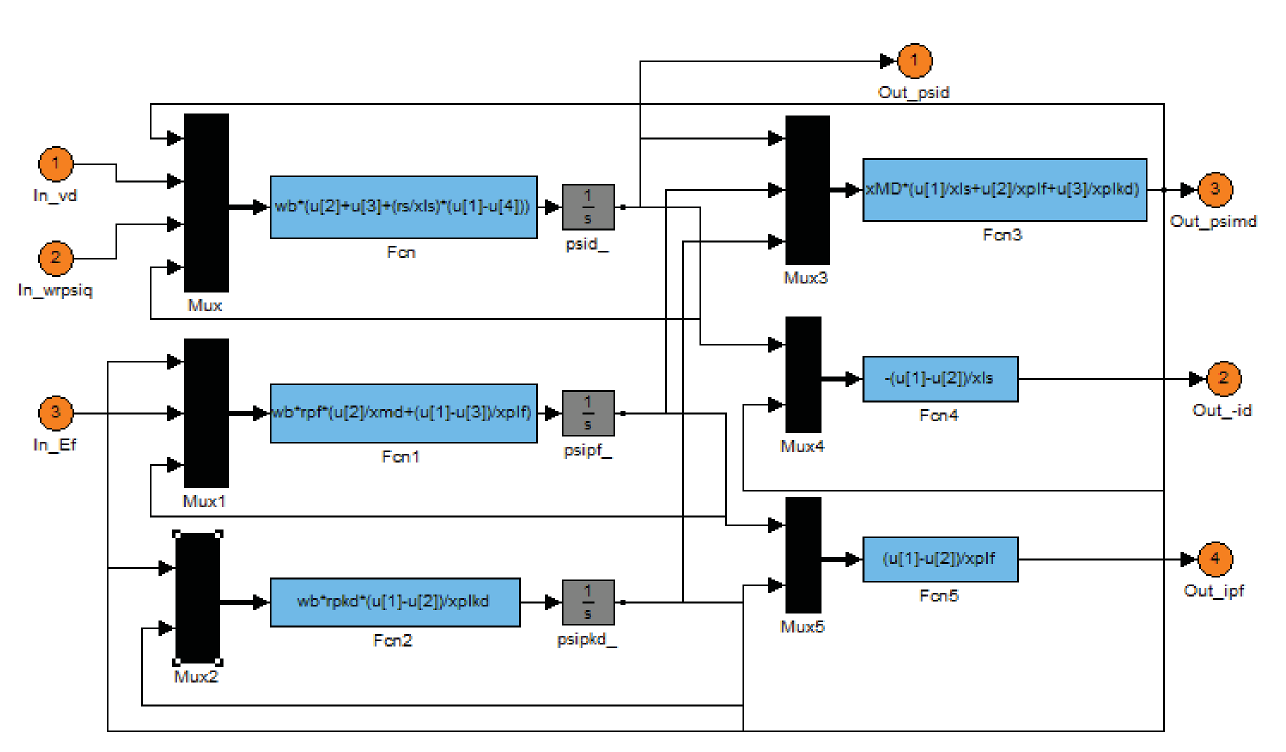


1
vq
psiq
-iq
T
q_cct
1
iq_gen
prod1
prod2
2
vd
3
Ef
psid
-id
T1
d_cct
5
ipf
6
id_gen
4
Tmech
Rotor
T2
2
delta
3
wr/wb
4
Tem

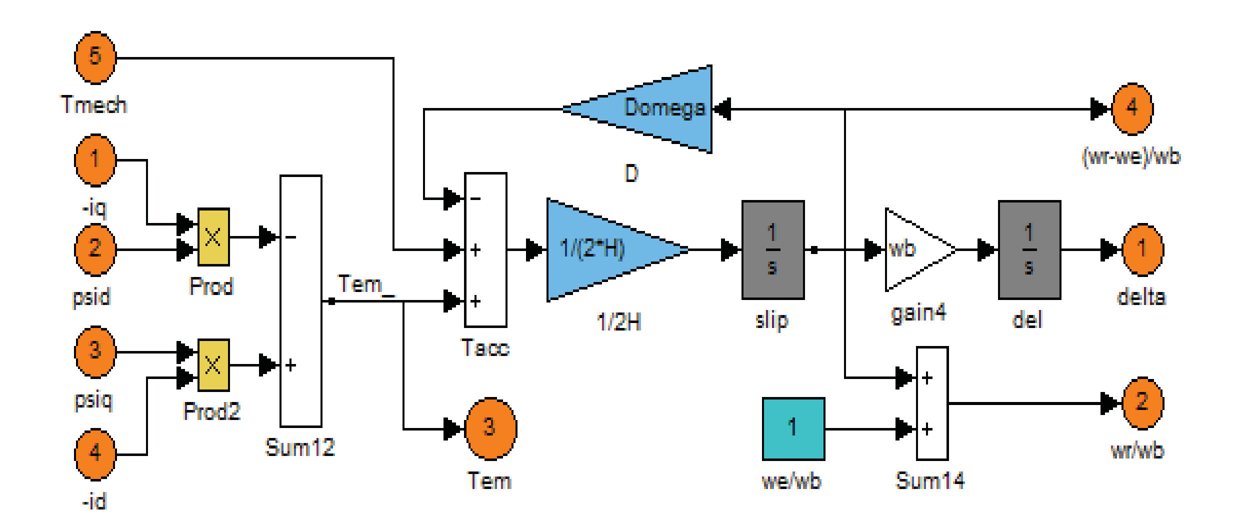
q_cct


1
Out_psiq
1
In_vq
2
In_wrpsid
Mux
wb*(u[2]-u[3]+(rs/xls)*(u[1]-u[4]))
Fcn
1
s
psiq_
Mux3
xMQ*(u[1]/xls+u[2]/xplkq)
Fcn3
3
Out_psimq
Mux2
wb*rpkq*(u[1]-u[2])/xplkq
Fcn2
1
s
psipkq_
Mux4
-(u[1]-u[2])/xls
Fcn4
2
Out_-iq

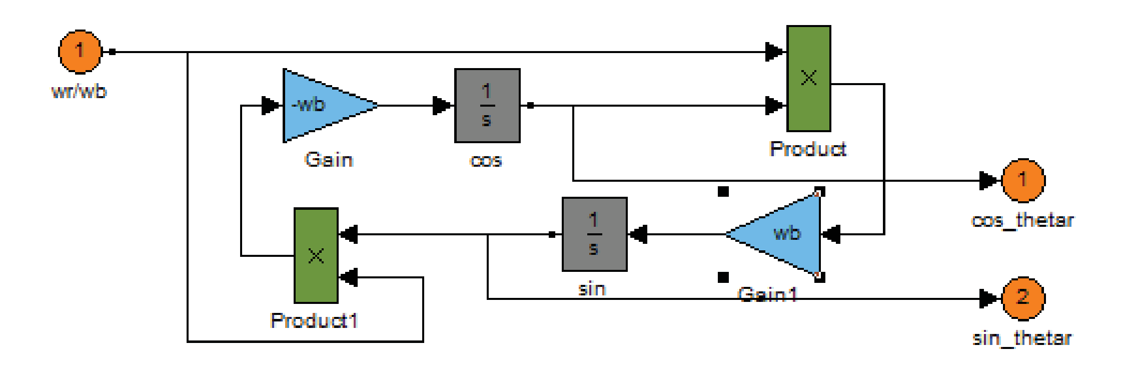
d_cct


1
Out_psid
1
In_vd
2
In_wrpsiq
Mux
wb*(u[2]+u[3]+(rs/xls)*(u[1]-u[4]))
Fcn
1
s
psid_
Mux3
xMD*(u[1]/xls+u[2]/xplf+u[3]/xplkd)
Fcn3
3
Out_psimd
3
In_Ef
Mux1
wb*rpf*(u[2]/xmd+(u[1]-u[3])/xplf)
Fcn1
1
s
psipf_
Mux4
-(u[1]-u[2])/xls
Fcn4
2
Out_-id
Mux2
wb*rpkd*(u[1]-u[2])/xplkd
Fcn2
1
s
psipkd_
Mux5
(u[1]-u[2])/xplf
Fcn5
4
Out_ipf

…Rotor…


5
Tmech
1
-iq
2
psid
3
psiq
4
-id
Prod
Prod2
Sum12
Tem_
Tacc
3
Tem
Domega
D
1/(2*H)
1/2H
1
s
slip
wb
gain4
1
s
del
4
(wr-we)/wb
1
delta
1
we/wb
Sum14
2
wr/wb

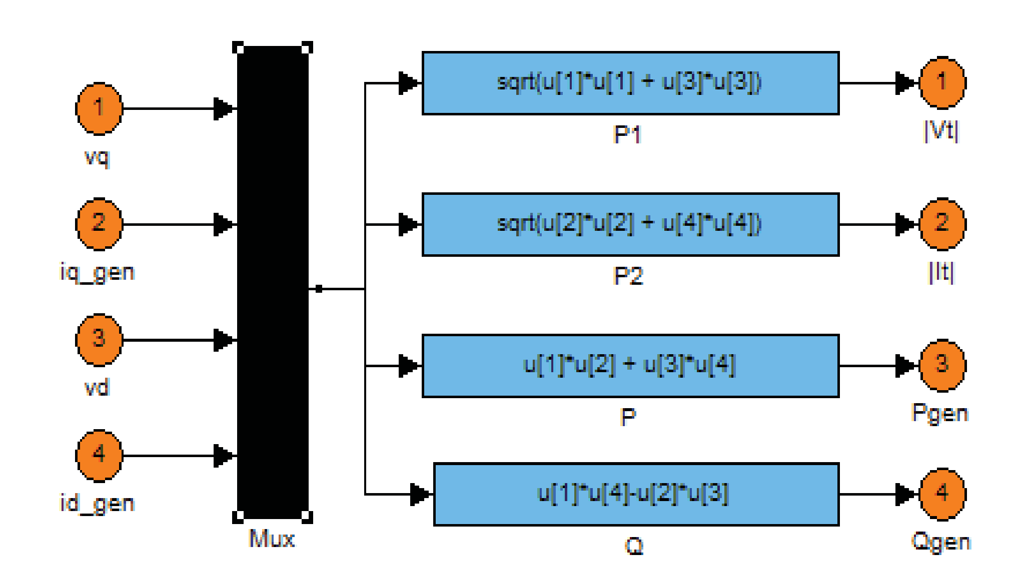
…3…


1
wr/wb
-wb
Gain
1
s
cos
Product
1
cos_thetar
1
s
sin
wb
Gain1
Product1
2
sin_thetar

...4...

qdr2qds
qds2abc
4
cos_thetar
5
sin_thetar
1
-iq
2
-id
Mux
3
i0
u[1]*u[3] + u[2]*u[4]
qdr2qs
u[1]*u[4] - u[2]*u[3]
qdr2ds
Mux1
u[1] + u[3]
qds2a
-u[1]/2 -sqrt(3)*u[2]/2 + u[3]
qds2b
-u[1]/2 +sqrt(3)*u[2]/2 + u[3]
qds2c
1
ia
2
ib
3
ic

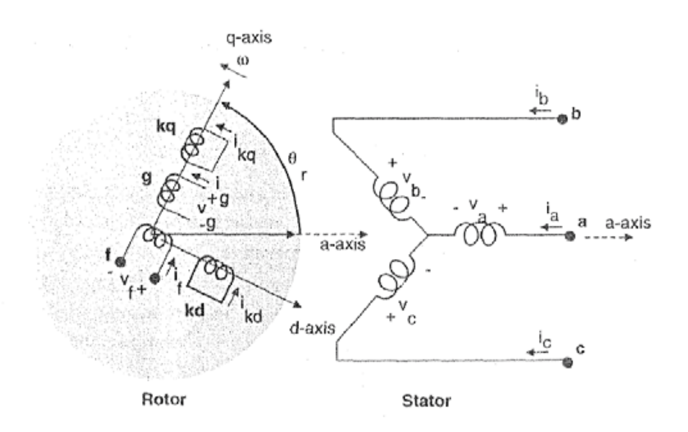
...5...



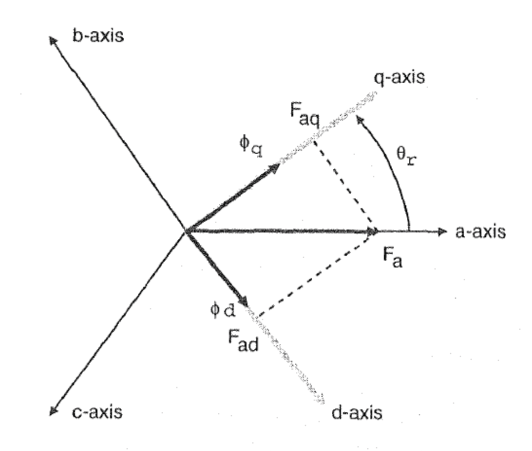
٤,١٥ نمذجة ومحاكاة المحرك التزامنية ذو المغناطيس الدائم

١,٤,١٥ الموديل الرياضي

الشكل التالي يبين الدارة الممثلة لموديول آلة تزامنية مثالية :



قبل اشتقاق المعادلات الرياضية للدارة السابقة ، سنأخذ نظرة موجزة لتغير المحارضات مع موضع الدائر. بشكل عام فإن المركبات على المحورين (dq) لا تكون متماثلة. حيث يتم توجيه القوة المحركة المغناطيسية لملفات الدائر دوماً على المحور العمودي أو المباشر ، وبالنتيجة فإن جهة القوة المحركة المغناطيسية لملفات الثابت تكون متغيرة مع معامل الاستطاعة. لنأخذ مثال مبسط لإيجاد مركبات القوة المحركة المغناطيسية للطور الأول a على المحورين المتعامدين. القوة المحركة المغناطيسية F_a تولد مركبات السيالة على المحورين $\varphi_d = P_d.F_a.\sin(\theta_r)$ ، $\varphi_q = P_q.F_a.\cos(\theta_r)$.



مركبات السيالة التسريبية للطور الأول a تكتب :

$$\lambda_{aa} = N_s\left(\varphi_d.\sin(\theta_r) + \varphi_q.co\,s(\theta_r)\right) \quad Wb.turn$$

$$= N_s.F_a\left(P_d.sin^2(\theta_r) + P_d.cos^2(\theta_r)\right)$$

$$= N_s.F_a\left(\frac{P_d + P_q}{2} - \frac{P_d - P_q}{2}\cos(2\theta_r)\right)$$

وبشكل مشابه فإن مركبة السيالة التسريبية للطور الثاني b تكون فقط مزاحة عن السابقة بزاوية ١٢٠ درجة وتكتب :

$$\lambda_{ba} = N_s.F_a\left(P_d.sin(\theta_r).\sin(\theta_r - \frac{2\pi}{3}) + P_d.cos(\theta_r)\cos(\theta_r - \frac{2\pi}{3})\right)$$

$$= N_s.F_a\left(-\frac{P_d + P_q}{4} - \frac{P_d - P_q}{2}\cos2\left(\theta_r - \frac{\pi}{3}\right)\right)$$

بالاعتماد على تبعية λ_{aa} لزاوية الدائر θ_r يمكن أن نستنتج المحارضات الذاتية للثابت للطور الأول كما يلي :

$$L_{aa} = L_0 - L_{ms}.\cos(2\theta_r) \qquad H$$

وبالمثل فإن المحارضات الذاتية للطور الثاني L_{bb} والثالث L_{cc} مشابهة لـ λ_{aa} وبتبديل θ_r والإزاحة الطور كما يلي $(\theta_r - \frac{4\pi}{3})$ and $\left(\theta_r - \frac{2\pi}{3}\right)$. وبالمثل يمكن ان نستنتج المحارضات التبادلية بين الأطوار للثابت بالصيغة :

$$L_{ab} = L_{ba} = -\frac{L_0}{2} - L_{ms}cos2\left(\theta_r - \frac{\pi}{3}\right) \qquad H$$

وبالمثل فإن المحارضات التبادلية L_{bc}, L_{ca} يمكن أن نحصل عليها بتبديل θ_r والإزاحة الطور كما يلي $\left(\theta_r - \frac{4\pi}{3}\right)$ and $\left(\theta_r - \frac{2\pi}{3}\right)$.

عند العمل كمحرك يتم تطبيق الجهد على كل من الملفات المبينة بالشكل ١,٣ وبشكل متوازن مع هبوط الجهد على المقاومة وكذلك $\frac{d\lambda}{dt}$. وبالتالي نكتب معادلات الجهد على كل من الثابت والدائر بالصيغة التالية :

$$\begin{bmatrix} v_s \\ v_r \end{bmatrix} = \begin{bmatrix} r_s & 0 \\ 0 & r_r \end{bmatrix} . \begin{bmatrix} i_s \\ i_r \end{bmatrix} + \frac{d}{dt}\begin{bmatrix} \Lambda_s \\ \Lambda_r \end{bmatrix} \qquad V$$

حيث :

$$v_s = [v_a, v_b, v_c]^t$$

$$v_r = \left[v_f, v_{kd}, v_g, v_{kq}\right]^t$$

$$i_s = [i_a\ i_b\ i_c]^t$$

$$i_r = [i_f\ i_{kd}\ i_g\ i_{kq}]^t$$

$$r_s = diag[r_a\, r_b\, r_c]$$

$$r_r = diag[r_f\ r_{kd}\ r_g\ r_{kq}]$$

$$\Lambda_s = [\lambda_a, \lambda_b, \lambda_c]^t$$

$$\Lambda_r = [\lambda_f, \lambda_{kd}, \lambda_g, \lambda_{kq}]^t$$

معادلات السيالة للثابت والدائر يعبر عنها بالمعادلتين :

$$\Lambda_s = L_{ss} i_s + L_{sr} i_r \qquad Wb.turn$$

$$\Lambda_r = [L_{sr}]^t i_s + L_r i_r$$

٢,٤,١٥ معادلات الآلة التزامنية على المحاور المتعامدة dq

$$v_q = r_s i_q + \frac{d\lambda_q}{dt} + \lambda_d \frac{d\theta_r}{dt} \qquad V$$

$$v_q = r_s i_d + \frac{d\lambda_d}{dt} - \lambda_q \frac{d\theta_r}{dt}$$

$$v_0 = r_s i_0 + \frac{d\lambda_0}{dt}$$

$$v'_f = r'_f i'_f + \frac{d\lambda'_f}{dt}$$

$$v'_{kd} = r'_{kd} i'_{kd} + \frac{d\lambda'_{kd}}{dt}$$

$$v'_g = r'_g i'_g + \frac{d\lambda'_g}{dt}$$

$$v'_{kq} = r'_{kq} i'_{kq} + \frac{d\lambda'_{kq}}{dt}$$

حيث تعطى السيالات التسربية بالعلاقات التالية :

$$\lambda_q = L_q i_q + L_{mq} i'_g + L_{mq} i'_{kq}$$

$$\lambda_d = L_d i_d + L_{md} i'_f + L_{md} i'_{kd}$$

$$\lambda_0 = L_{ls} i_0$$

$$\lambda'_f = L_{md} i_d + L_{md} i'_{kd} + L'_f i'_f$$

$$\lambda'_{kd} = L_{md} i_d + L_{md} i'_f + L'_{kdkd} i'_{kd}$$

$$\lambda'_g = L_{mq} i_q + L'_{gg} i'_g + L_{mq} i'_{kq}$$

$$\lambda'_{kq} = L_{mq} i_q + L_{mq} i'_g + L'_{kqkg} i'_{kq}$$

الشكل 5.3 يبين الدارة المكافئة للآلة التزامنية بالاعتماد على العلاقات بين الجهد والسيالة.

٣,٤,١٥ العزم الكهرطيسي

يتم الحصول على العزم الكهرطيسي في الآلة التزامنية من مركبة استطاعة الدخل بعد أن تعبر الثغرة الهوائية. حيث تعطى استطاعة دخل الآلة بالعلاقة :

$$P_{in} = v_a i_a + v_b i_b + v_c i_c + v_f i_f + v_g i_g \qquad W$$

وبالتعبير عن استطاعة الدخل على المحاور المتعامدة باعتبار أن $w_r = d\theta_r/dt$ نكتب :

$$P_{in} = \frac{3}{2}\left(v_q i_q + v_d i_d\right) + 3v_0 i_0 + v_f i_f + v_g i_g$$

$$= \frac{3}{2}\left(r_s\left(i_q^2 + i_d^2\right) + i_q \frac{d\lambda_q}{dt} + i_d \frac{d\lambda_d}{dt} + w_r\left(\lambda_d i_q - \lambda_q i_d\right)\right)$$

$$+ 3i_0^2 r_0 + 3i_0 \frac{d\lambda_0}{dt} + i_f^2 r_f + i_f \frac{d\lambda_f}{dt} + i_g^2 r_g + i_g \frac{d\lambda_g}{dt}$$

بإزالة الضياعات الأومية و التغير في الطاقة المغناطيسية من المعادلة الأخيرة نحصل على العلاقة التي تعبر عن الاستطاعة الكهرطيسية الناتجة :

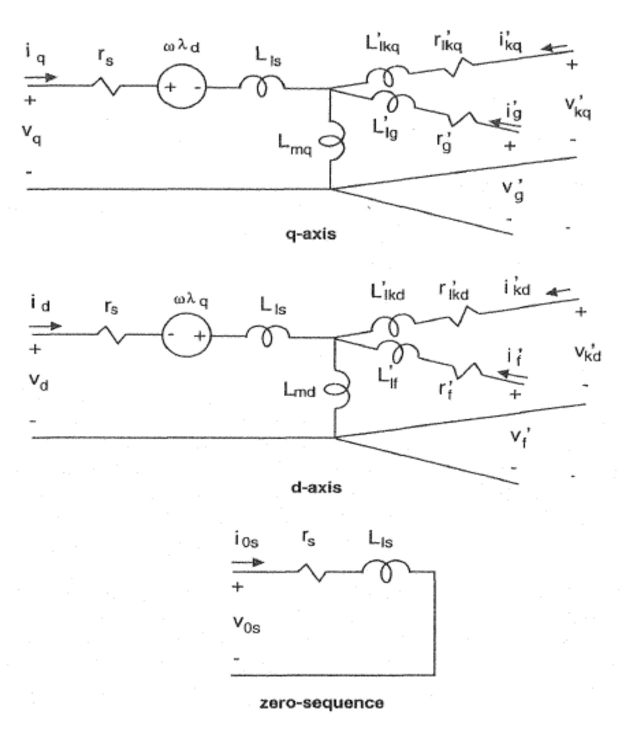
$$P_{em} = \frac{3}{2} w_r\left(\lambda_d i_q - \lambda_q i_d\right) \qquad W$$

من أجل آلة بعدد أقطاب P تصبح العلاقة كما يلي :

$$P_{em} = \frac{3}{2}\frac{P}{2} w_{rm}\left(\lambda_d i_q - \lambda_q i_d\right) \qquad W$$

وبتقسيم العلاقة الأخيرة على السرعة الميكانيكية نحصل على العزم الكهرطيسي المطلوب من أجل آلة تزامنية بعدد أقطاب P وذلك كما توضح العلاقة :

$$T_{em} = \frac{3}{2}\frac{P}{2}\left(\lambda_d i_q - \lambda_q i_d\right) \qquad N.m$$



٤,٤,١٥ التيارات كتابع للسيالات التبادلية

غالباً تتم محاكاة الآلات التزامنية بجعل السيالات التبادلية بين الملفات هي المتغيرة. نكتب معادلات السيالة التبادلية على المحاور المتعامدة كما يلي :

$$\lambda_{mq} = L_{mq}\left(i_q + i'_g + i'_{kq}\right) \qquad Wb.turn$$

$$\lambda_{md} = L_{md}(i_d + i'_f + i'_{kd})$$

ويمكن كتابة علاقات التيار بصيغة مبسطة كما يلي :

$$i_q = \frac{1}{L_{ls}}\left(\lambda_q - \lambda_{mq}\right) \qquad i_d = \frac{1}{L_{ls}}(\lambda_d - \lambda_{md})$$

$$i'_g = \frac{1}{L'_{lg}}(\lambda'_g - \lambda_{mq}) \qquad i'_f = \frac{1}{L'_{lf}}(\lambda'_f - \lambda_{md})$$

$$i'_{kq} = \frac{1}{L'_{lkq}}(\lambda'_{kq} - \lambda_{mq}) \qquad i'_{kd} = \frac{1}{L'_{lkd}}(\lambda'_{kd} - \lambda_{md})$$

بتعويض قيمة التيارات على المحور المباشر بالسيالة التبادلية نحصل على قيمة السيالة التبادلية كتابع للتيار

$$\lambda_{md} = \frac{L_{MD}}{L_{ls}}\lambda_d + \frac{L_{MD}}{L'_{lf}}\lambda'_f + \frac{L_{MD}}{L'_{lkd}}\lambda'_{kd} \qquad Wb.turn$$

حيث :

$$\frac{1}{L_{MD}} = \frac{1}{L_{ls}} + \frac{1}{L'_{lf}} + \frac{1}{L'_{lkd}} + \frac{1}{L_{md}}$$

٥,٤,١٥ العمل في الحالة المستقرة

$$v_a = V_m \cos(w_e t) \qquad V$$

$$v_b = V_m \cos\left(w_e t - \frac{2\pi}{3}\right)$$

$$v_c = V_m \cos\left(w_e t - \frac{4\pi}{3}\right)$$

$$i_a = I_m \cos(w_e t + \varphi)$$

$$i_b = I_m \cos\left(w_e t + \varphi - \frac{2\pi}{3}\right)$$

$$i_c = I_m \cos\left(w_e t + \varphi - \frac{4\pi}{3}\right)$$

من الواضح من معادلات الجهد والتيارات أن زاوية معامل الاستطاعة φ تكون موجبة عندما يكون معامل الاستطاعة متقدم و تكون سالبة عندما يكون معامل الاستطاعة متأخر.

في هذه المرحلة ، لا نستطيع معرفة جهة محور الدائر q_r مع محور السيالة التي تدور بالسرعة التزامنية q_e. وبما أن الدائر يدور بالسرعة التزامنية في الحالة المستقرة ، فإننا نعلم الزاوية بين المحورين (q_r, q_e) والتي ستكون قيمتها ثابتة ولا تتغير مع الزمن. ومن أجل تحديد موضع محور الدائر q_r ، نقوم بنسب كل من الجهد والتيار من المحاور ثلاثية الطور للسيالة الدائرة بالسرعة التزامنية (التي بيناها بالفصل السابق) وتكون مركبات كلاً من الجهد والتيار على المحاور المتعامدة dq منسوباً للسيالة الدائرة بالسرعة التزامنية كما يلي :

$$v_q^e - jv_d^e = V_m + j0 = V_m e^{j0}$$

$$i_q^e - ji_d^e = I_m\cos(\varphi) + jI_m\sin(\varphi) = I_m e^{j\varphi}$$

نلاحظ مركبات الجهد والتيار على كل من المحاور المتعامدة dq تكون ذات قيمة ثابتة بينما تكون قيمة المركبة الصفرية معدومة عن توازن الاطوار الثلاثة.

١٥,٥,٤,١ معادلات الحالة المستقرة للثابت

عادة فقط ملفات الحقل تتم تغذيتها من منبع خارجي ، أما باقي ملفات الدائر فلا يوجد فيها أي دخل من منبع خارجي وبالتالي $v'_{kd} = v'_g = v'_{kq} = 0$. في الحالة المستقرة فإن الدائر يدور بالسرعة التزامنية وبالتالي $w_r(t) = d\theta_r(t)/dt = w_e$. والسرعة النسبية للدائر منسوباً للدائر التزامني (السيالة) ينتج عنها حقل معدوم في الثغرة الهوائية. ولن يكون هناك أي توترات ناتجة عن السرعة في ملفات الدائر ولذلك يكون $i'_f = v'_f/r'_f$ ، وتيارات الدائر الأخرى $i'_g, i'_{kd}, i'_{kq} = 0$. وبما أن كل من تيارات الثابت والدائر ثابتة، فإن السيالات التسربية λ_d, λ_q ستكون أيضاً ثابتة وتغييرات السيالة مع الزمن ستكون معدومة. ولذلك في حالة العمل المستقرة فإن الجهود على المحورين المتعامدين dq لملفات الثابت منسوباً للدائر dq ستختصر لتصبح :

$$v_q = r_s i_q + w_e L_d i_d + E_f \qquad V$$

$$v_d = r_s i_d - w_e L_q i_q$$

حيث E_f تشير لمنبع التهييج في الحالة المستقرة لحانب الثابت :

$$E_f = w_e L_{md} \left(\frac{v'_f}{r'_f} \right) \qquad V$$

١٥,٥,٤,٢ تحديد موضع الدائر

سنعرف الزاوية $\delta(t)$ بين المحورين q_e, q_r :

$$\delta(t) = \theta_r(t) - \theta_e(t) \qquad electrical.rad.$$

$$= \int_0^t (w_r(t) - w_r)dt + \theta_r(0) - \theta_e(0)$$

حيث :

θ_r : الزاوية بين المحور q_r والطور الأول a.

θ_e : الزاوية بين المحور q_e والطور الأول a.

في الحالة المستقرة عند الدوران بالسرعة التزامنية $w_r(t) = w_e$ فإن الزاوية δ ستكون ثابتة.

١٥,٥,٤,٣طويلة الزمن (*Time Phasors*) والأشعة الفراغية (*Space Vectors*)

تيار الثابت بالصيغة الفراغية على المحورين المتعامدين dq يعبر عنه بالعلاقة :

$$\overrightarrow{I_s^s} = i_q^s - ji_d^s = \frac{2}{3}(i_a + ai_b + a^2 i_c) \qquad A$$

حيث s تشير إلى المتغيرات على المحاور dq ، بإجراء التعديل :

$$a = e^{j\frac{2\pi}{3}} \quad \text{و} \quad \cos(w_e t + \varphi) = \frac{\left(e^{j(w_e t+\varphi)} + e^{-j(w_e t+\varphi)}\right)}{2}$$

والتبديل بالمعادلة السابقة نجد :

$$i_q^s - ji_d^s = I_m e^{j\varphi} e^{jw_e t} \qquad A$$

والقيمة اللحظية لطويلة تيار الطور الأول $\tilde{I}_a$

$$\tilde{I}_a = \frac{I_m}{\sqrt{2}} e^{j\varphi}$$

وللتعبير عن ذلك كطويلة وزاوية نكتب :

$$\vec{I}_s^s = i_q^s - ji_d^s = \sqrt{2}\tilde{I}_a e^{jw_e t} \qquad A$$

العلاقة بين التيار اللحظي للطور الأول و طويلة التيار كما يلي :

$$i_a = \mathfrak{R}\left[\sqrt{2}\tilde{I}_a e^{jw_e t}\right] \qquad A$$

ونفس العلاقات تكتب بالنسبة للتوتر ، وتكون القيم اللحظية للتوتر على المحورين المتعامدين :

$$\overrightarrow{V_q} = r_s \overrightarrow{I_q} + w_e L_d \overrightarrow{I_d} + \overrightarrow{E_f}$$

$$\overrightarrow{V_d} = r_s \overrightarrow{I_d} - w_e L_q \overrightarrow{I_q}$$

٤,٤,٥,١٥ العزم

$$T_{em} = \frac{P_{em}}{w_{sm}} = \left(\frac{2}{Pw_e}\right) P_{em} \qquad N.m$$

$$= 3\left(\frac{2}{Pw_e}\right)\{E_f I_q + w_e(L_d - L_q) I_d I_q\}$$

$$T_{em} = -3\left(\frac{2}{P.w_e}\right)\{\frac{V_a.E_f}{X_d}.\sin(\delta) + \frac{V_a^2}{2}\left(\frac{1}{X_q} - \frac{1}{X_d}\right).\sin(2\delta)\}$$

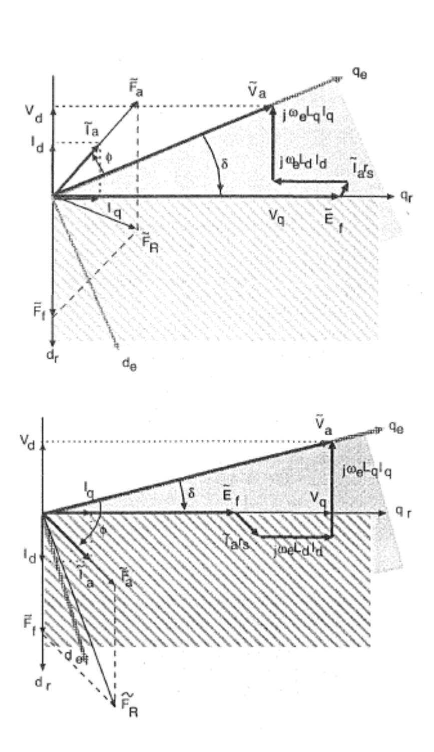
٥,٤,٥,١٥ المخطط الشعاعي لحالات عمل المحرك التزامني

الشكل ٤,٥ يبين المخطط الشعاعي للمحرك التزامني عند معامل استطاعة متقدم ومتأخر وذلك بالاعتماد على العلاقات التالية :

$$\overrightarrow{V_q} = r_s\overrightarrow{I_q} + w_s L_d \overrightarrow{I_d} + \overrightarrow{E_f} \qquad V$$

$$\overrightarrow{V_d} = r_s\overrightarrow{I_d} - w_e L_q \overrightarrow{I_q}$$

في المحركات التزامنية ذات المغناطيس الدائم يمكن تعويض منبع التهييج الخارجي المستمر بالمغناطيس الدائم الذي تم وضعه بدلاً من ملفات الدائر(ملفات الحقل). إن هذا التبديل سوف يؤدي لبنية أبسط وكذلك وزن أخف وحجم أصغر معطيةً نفس الأداء ، مع تخفيض الضياعات. وتتمثل مساوئ هذا النوع من المحركات بالسعر المرتفع للمواد المغناطيسية المستخدمة ، وأيضاً إحدى أهم المساوئ أن خواص المواد المغناطيسية المستخدمة قد تتغير مع الزمن. إن اختيار المواد المغناطيسية المستخدمة تنعكس مباشرة على أداء المحرك، وزنه، حجمه، أدائه، والتكلفة الاقتصادية للمواد المستخدمة ، وبالتالي تكلفة المحرك. يتم إقلاع هذا النوع من المحركات بوصلها على الشبكة مباشرة لتقلع كمحرك تزامني ذو قفص سنجابي دائر بمساعدة منبع التردد الثابت. إن مركبة العزم التحريضي الناتجة عن الإقلاع أصبحت عزوم نبضية تنشأ عن الدائر المغناطيسي أثناء الإقلاع. عند تغذية المحرك من مبدلة إلكترونية فمن الممكن أن تقلع كمحرك تزامني ذو قفص سنجابي دائر أو لا حيث أن تردد المبدلة يعمل على وصول سرعة الدائر للسرعة التزامنية.



٦,١٥ معادلات المحرك التزامني ذو المغناطيس الدائم

معادلات الجهد :

$$v_q = r_s . i_q + \frac{d\lambda_q}{dt} + \lambda_d . \frac{d\theta_r}{dt}$$

$$v_d = r_s . i_d + \frac{d\lambda_d}{dt} - \lambda_q . \frac{d\theta_r}{dt}$$

$$v_0 = r_s . i_0 + \frac{d\lambda_0}{dt}$$

$$0 = r'_{kd} . i'_{kd} + \frac{d\lambda'_{kd}}{dt}$$

$$0 = r'_{kq} . i'_{kq} + \frac{d\lambda'_{kq}}{dt}$$

السيالة التسربية :

$$\lambda_q = L_q . i_q + L_{mq} . i'_{kq} \qquad Wb.turn$$

$$\lambda_d = L_d . i_d + L_{md} . i'_{kd} + \underbrace{L_{md} . i'_m}_{\lambda'_m}$$

$$\lambda_0 = L_{ls} . i_0$$

$$\lambda'_{kq} = L_{mq} . i_q + L'_{kq} . i'_{kq}$$

$$\lambda'_{kd} = L_{md} . i_d + L'_{kd} . i'_{kd} + L_{md} . i'_m$$

العزم الكهرومغناطيسي :

$$T_{em} = \frac{3}{2}\frac{P}{2}\left(\lambda_d i_q - \lambda_q i_d\right) \qquad N.m$$

$$T_{em} = \frac{3}{2}\frac{P}{2}\left(L_d - L_q\right) i_d . i_q + \frac{3}{2}\frac{P}{2}\left(L_{md} . i'_{kd} i_q - L_{mq} . i'_{kq} i_d\right) + \frac{3}{2}\frac{P}{2} L_{md} . i'_m . i_q$$

المعادلة الأخيرة تبين أن العزم الكهرومغناطيسي ينفصل لثلاث مركبات أساسية كما هو مبين.

يعبر عن السيالة التبادلية على المحورين المتعامدين بالمعادلات :

$$\lambda_{mq} = L_{mq}\left(i_q + i'_{kq}\right) \qquad Wb.turn$$

$$\lambda_{md} = L_{md}(i_d + i'_{kq} + i'_m)$$

يعبر عن التيارات المارة بالملفات بالمعادلات :

$$i_q = \frac{\lambda_q - \lambda_{mq}}{L_{ls}}$$

$$i_d = \frac{\lambda_d - \lambda_{md}}{L_{ls}}$$

$$i'_{kd} = \frac{\lambda'_{kd} - \lambda_{md}}{L'_{lkd}}$$

$$i'_{kq} = \frac{\lambda'_{kq} - \lambda_{mq}}{L'_{lkq}}$$

بتعويض معادلات التيارات في معادلات السيالات وتبسيط العلاقة نجد :

$$\lambda_{md} = L_{MD}(\frac{\lambda_d}{L_{ls}} + \frac{\lambda'_{kd}}{L'_{lkd}} + i'_m) \qquad Wb.turn$$

حيث :

$$\frac{1}{L_{MD}} = \frac{1}{L_{ls}} + \frac{1}{L'_{lkd}} + \frac{1}{L_{md}}$$

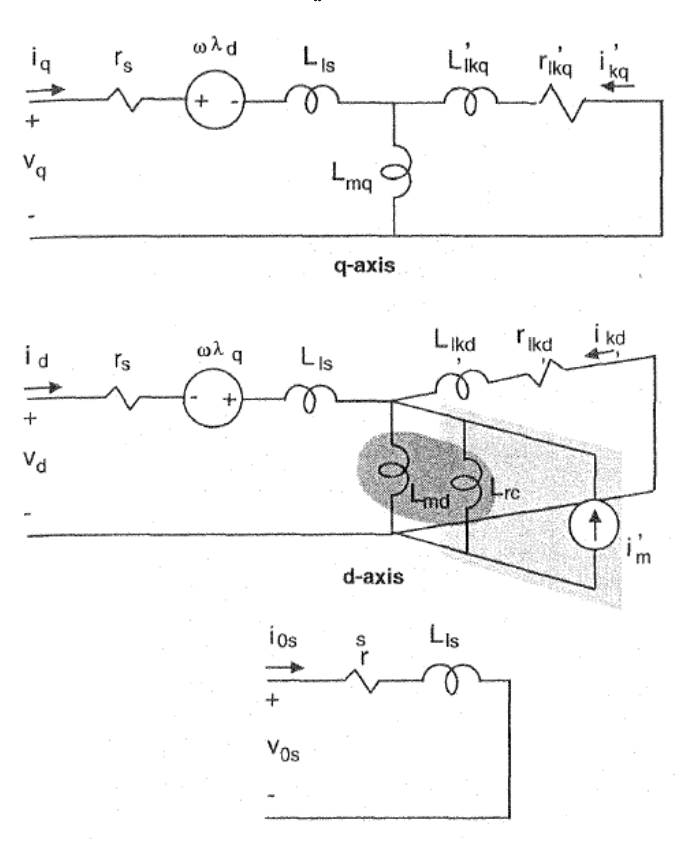
وبالمثل معادلات λ_{mq}, L_{mq} على المحور العمودي.

إن قيمة عزم المحرك الناتج عن المغناطيس الدائم كتابع للقيمة اللحظية للجهد على طرف الآلة يعبر عنه بالعلاقة التالية :

$$T_{em} = -3(\frac{2}{P.w_e})\{\frac{V_a.E_m}{X_d}.\sin(\delta) + \frac{V_a^2}{2}\left(\frac{1}{X_q} - \frac{1}{X_d}\right).\sin(2\delta)\}$$

إن المحرك التزامني ذو المغناطيس الدائم يكون ذو أداء عالي عند إقلاعه المباشر وتغذيته من منبع جهد ذي تردد ثابت. إن المغناطيس الدائم الموجود في الجزء الدائر للمحرك يؤمن التهييج المتزامن والقفص الدائر يؤمن العزم التحريضي للعزم عند الإقلاع. إن الاختلاف في النفاذية المغناطيسية بين المغناطيس ونواة الدائر ينشأ عنه عزم مقاوم عند السرعة التزامنية. عند الدوران بالسرعة

اللاتزامنية ، فإن منبع التهييج الذي هو في حالتنا المدروسة المغناطيس الدائم سيسبب عزوم نبضية. عندما يكون الحقل المغناطيس الناتج عن المغناطيس الدائم قوي جداً ، فإن المحرك قد يفشل في الدوران بالسرعة التزامنية وذلك بسبب العزوم النبضية الزائدة الناتجة من تهييج المغناطيس الدوار.
إن الهدف من محاكاة المحرك التزامني ذو المغناطيس الدائم هو دراسة المحرك عند إقلاعه من السكون والاستفادة من نموذج المحاكاة من أجل دراسة سلوك مركبات العزم أثناء الإقلاع. تتم تغذية المحرك من منبع جهد ذو تردد ثابت. والجدول التالي يبين المعادلات اللازمة للمحاكاة.



٧,١٥ النموذج الرياضي اللازم للمحاكاة

الانتقال من نموذج المحاور الثلاثية (abc) إلى المحورين المتعامدين (dq) يتم وفق المعادلات التالية :

$$v_q^s = \frac{2}{3}v_a - \frac{1}{3}v_b - \frac{1}{3}v_c$$

$$v_d^s = \frac{1}{\sqrt{3}}(v_c - v_b)$$

$$v_o = \frac{1}{3}(v_a + v_b + v_c)$$

$$v_q = v_q^s \cos\theta_r(t) - v_d^s \sin\theta_r(t)$$

$$v_d = v_q^s sin\theta_r(t) + v_d^s cos\theta_r(t)$$

حيث : $\theta_r(t) = \int_0^t w_r(t)\, dt + \theta_r(0)$ electrical.radian.

أو يمكن إجراء التحويل بشكل آخر بخطوة واحدة وفق المعادلات التالية :

$$v_q = \frac{2}{3}\{v_a \cos\theta_r(t) + v_b \cos(\theta_r(t) - \frac{2\pi}{3}) + v_c \cos(\theta_r(t) + \frac{2\pi}{3})\}$$

$$v_d = \frac{2}{3}\{v_a sin\theta_r(t) + v_b sin(\theta_r(t) - \frac{2\pi}{3}) + v_c sin(\theta_r(t) + \frac{2\pi}{3})\}$$

$$v_o = \frac{1}{3}(v_a + v_b + v_c)$$

وللتعبير عن معادلات الجهد على المحاور المتعامدة (dq) كتكامل للسيالة التسربية للملفات ، فإن المعادلات السابقة والتي تصف جهد الثابت يمكن أن تحول لكي تحل معادلات السيالة التسربية للملفات. وفي حالة الآلة بملف حقل واحد فقط على المحور المباشر وزوج من ملفات الإخماد على المحورين المتعامدين ، فإن معادلات السيالة على المحاور المتعامدة تصبح :

$$\psi_d = w_b \int \{(v_q - \frac{w_r}{w_b}\psi_d + \frac{r_s}{x_{ls}}(\psi_{mq} - \psi_q)\}dt$$

$$\psi_q = w_b \int \{(v_d + \frac{w_r}{w_b}\psi_q + \frac{r_s}{x_{ls}}(\psi_{md} - \psi_d)\}dt$$

$$\psi_0 = w_b \int (v_0 + \frac{r_s}{x_{ls}}\psi_0)dt$$

$$\psi'_{kq} = \frac{w_b r'_{kq}}{x'_{lkq}} \int (\psi_{mq} - \psi'_{kq})\, dt$$

$$\psi'_{kd} = \frac{w_b r'_{kd}}{x'_{lkd}} \int (\psi_{md} - \psi'_{kd})\, dt$$

حيث :

سنعبر عن السيالة التبادلية كتابع للسيالة الكلية التسربية في الملفات بحيث تكون المعادلات:

$$\psi_{mq} = x_{MQ}(\frac{\psi_q}{x_{ls}} + \frac{\psi'_{kq}}{x'_{lkq}})$$

$$\psi_{md} = x_{MD}(\frac{\psi_d}{x_{ls}} + \frac{\psi'_{kd}}{x'_{lkd}} + i'_m)$$

حيث :

$$\frac{1}{x_{MQ}} = \frac{1}{x_{mq}} + \frac{1}{x'_{lkq}} + \frac{1}{x_{ls}}$$

$$\frac{1}{x_{MD}} = \frac{1}{x_{md}} + \frac{1}{x'_{lkd}} + \frac{1}{x'_{lf}} + \frac{1}{x_{ls}}$$

وبعد الحصول على قيم السيالة التسربية للملفات والسيالة التشابكية التسربية على المحورين المتعامدين يمكن تحديد تيارات الملفات حسب المعادلات التالية :

$$i_q = \frac{\psi_q - \psi_{mq}}{x_{ls}}$$

$$i_d = \frac{\psi_d - \psi_{md}}{x_{ls}}$$

$$i'_{kd} = \frac{\psi'_{kd} - \psi_{md}}{x'_{lkd}}$$

$$i'_{kq} = \frac{\psi'_{kq} - \psi_{mq}}{x'_{lkq}}$$

التيارات على المحاور المتعامدة يمكن الحصول على القيم المكافئة لها على نموذج المحاور الثلاثية وفق المعادلات التالية :

$$i_q^s = i_q cos\theta_r(t) + i_d sin\theta_r(t)$$

$$i_d^s = -i_d sin\theta_r(t) + i_d cos\theta_r(t)$$

$$i_a = i_q^s + i_0$$

$$i_b = -\frac{1}{2} i_q^s - \frac{1}{\sqrt{3}} i_d^s + i_0$$

$$i_c = -\frac{1}{2} i_q^s + \frac{1}{\sqrt{3}} i_d^s + i_0$$

العزم الكهروميكانيكي (حركة الدائر) الناتج عن ألة تحوي عدد من الأقطاب محدد في حال العمل كمحرك تعطي بالعلاقة :

$$T_{em} = \left(\psi_d i_q - \psi_q i_d\right) \qquad pu$$

$$T_{em} = \left(x_d - x_q\right) i_d . i_q + \left(x_{md} . i'_{kd} i_q - x_{mq} . i'_{kq} i_d\right) + x_{md} . i'_m . i_q$$

$$T_{em(pu)} + T_{mech(pu)} - T_{damp(pu)} = 2H \frac{d\left({w_r - w_e}/{w_b}\right)}{dt}$$

زاوية الدائر تعرف كما يلي :

$$\delta(t) = \theta_r(t) - \theta_e(t)$$

$$= w_b \int_0^t \{w_r(t) - w_e\} dt + \theta_r(0) - \theta_e(0)$$

θ_r: زاوية الدائر بالنسبة للمحور العمودي أي في المجال المرجعي للدائر

θ_e: زاوية السيالة بالنسبة للمحور العمودي أي في المجال المرجعي للسيالة التزامنية

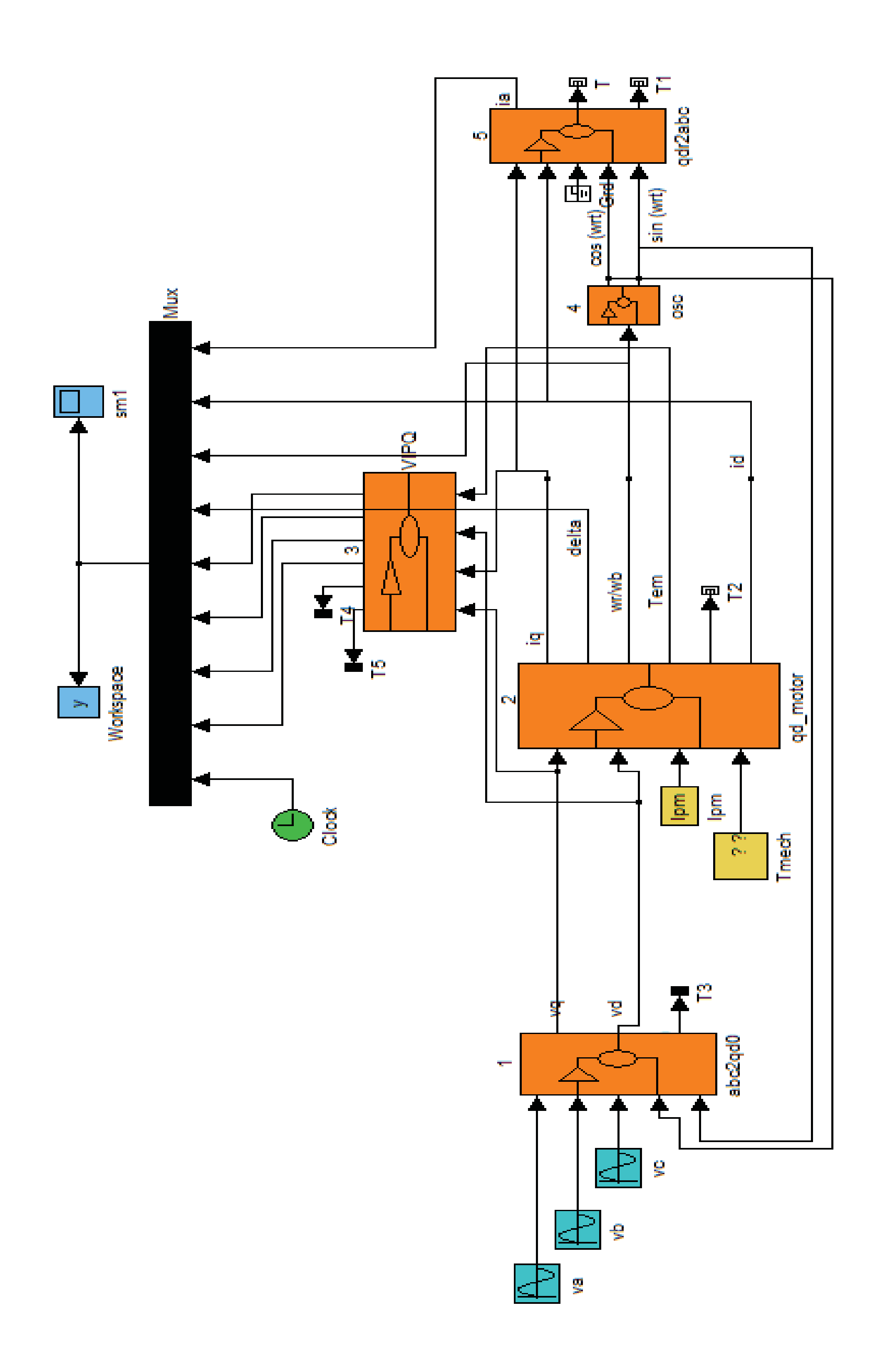
١,٧,١٥ التعبير بالقيم الواحدية لمعادلة العزم ومعادلة حركة الدائر :

عند دراسة نظم القدرة حيث يكون هناك العديد من المحولات وكذلك تجهيزات متعددة في الشبكة فلابد من أخذ نقطة أساس والعمل بالقيم الواحدية عندها (النسب للأساس).

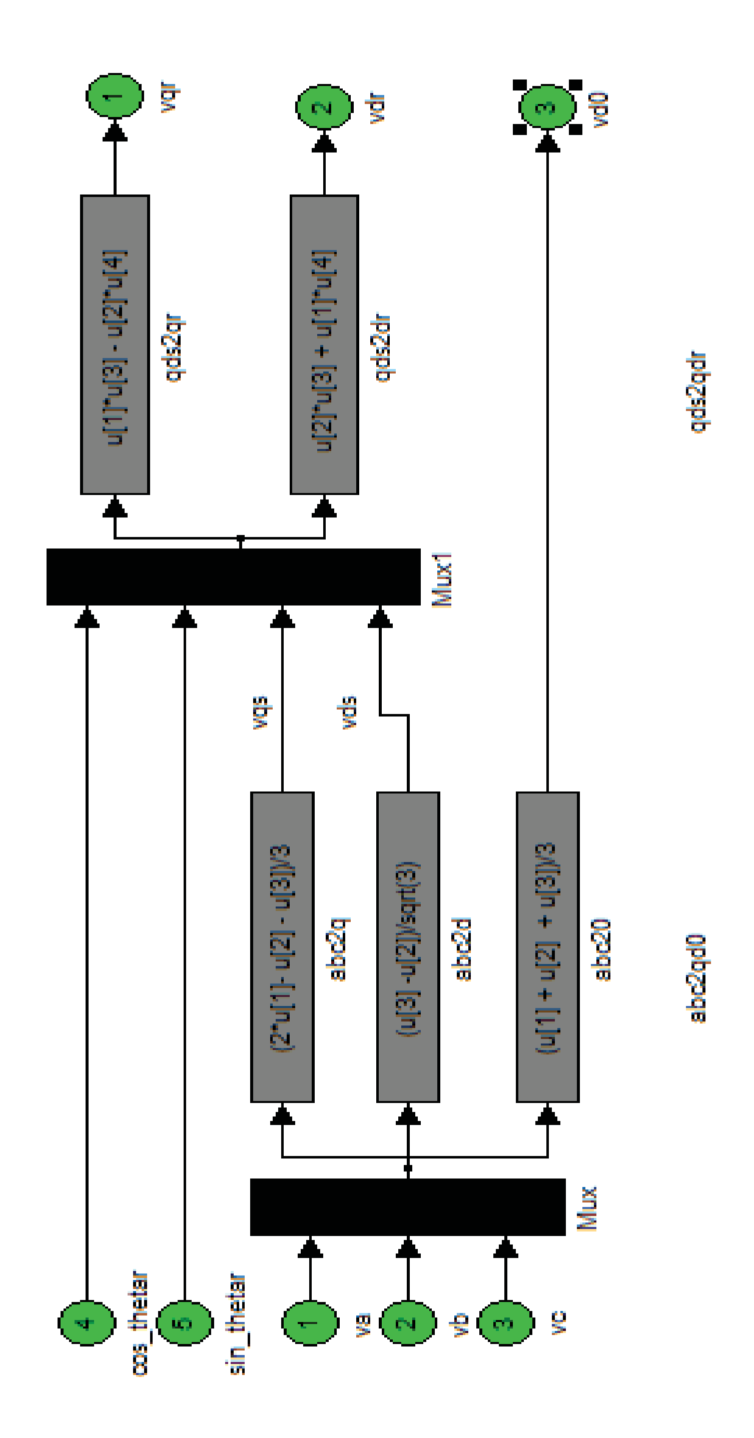
دراستنا تتضمن فقط ألة تزامنية واحدة ولذلك لن نستفيد من جميع نواحي استخدام النسب.

ممانعة الأساس $\quad Z_b = \frac{V_b}{I_b} \quad \Omega$

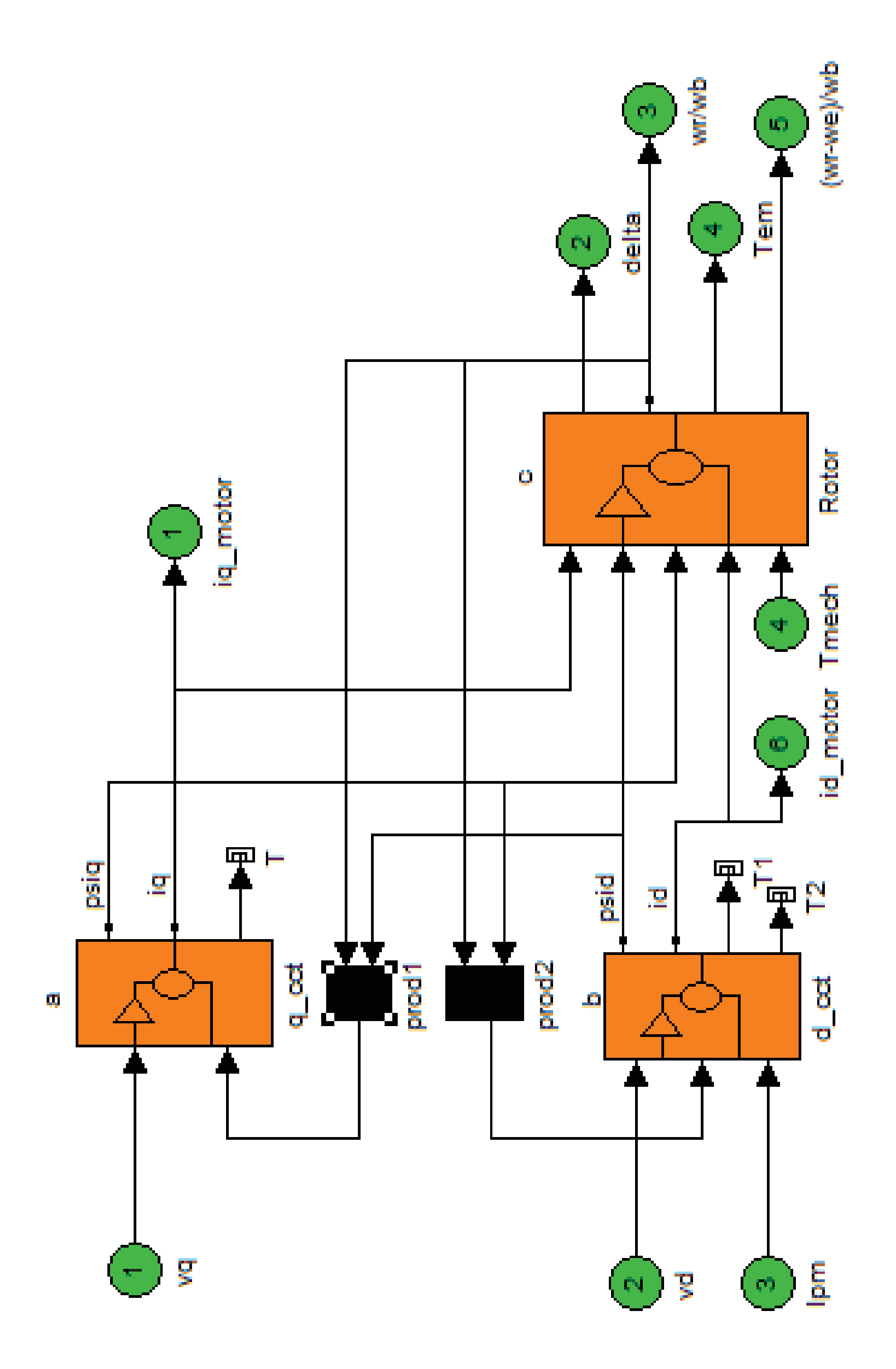
عزم الأساس $\quad T_b = \frac{S_b}{w_{bm}} \quad N.m$


Mux
sm1
y
Workspace
Clock
VPQ
3
T4
T5
iq
delta
wr/wb
Tem
T2
id
2
qd_motor
lpm
Tmech
5
ia
T
T1
qdr2abc
Gnd
cos (wrt)
sin (wrt)
4
osc
vq
vd
T3
1
abc2qd0
va
vb
vc

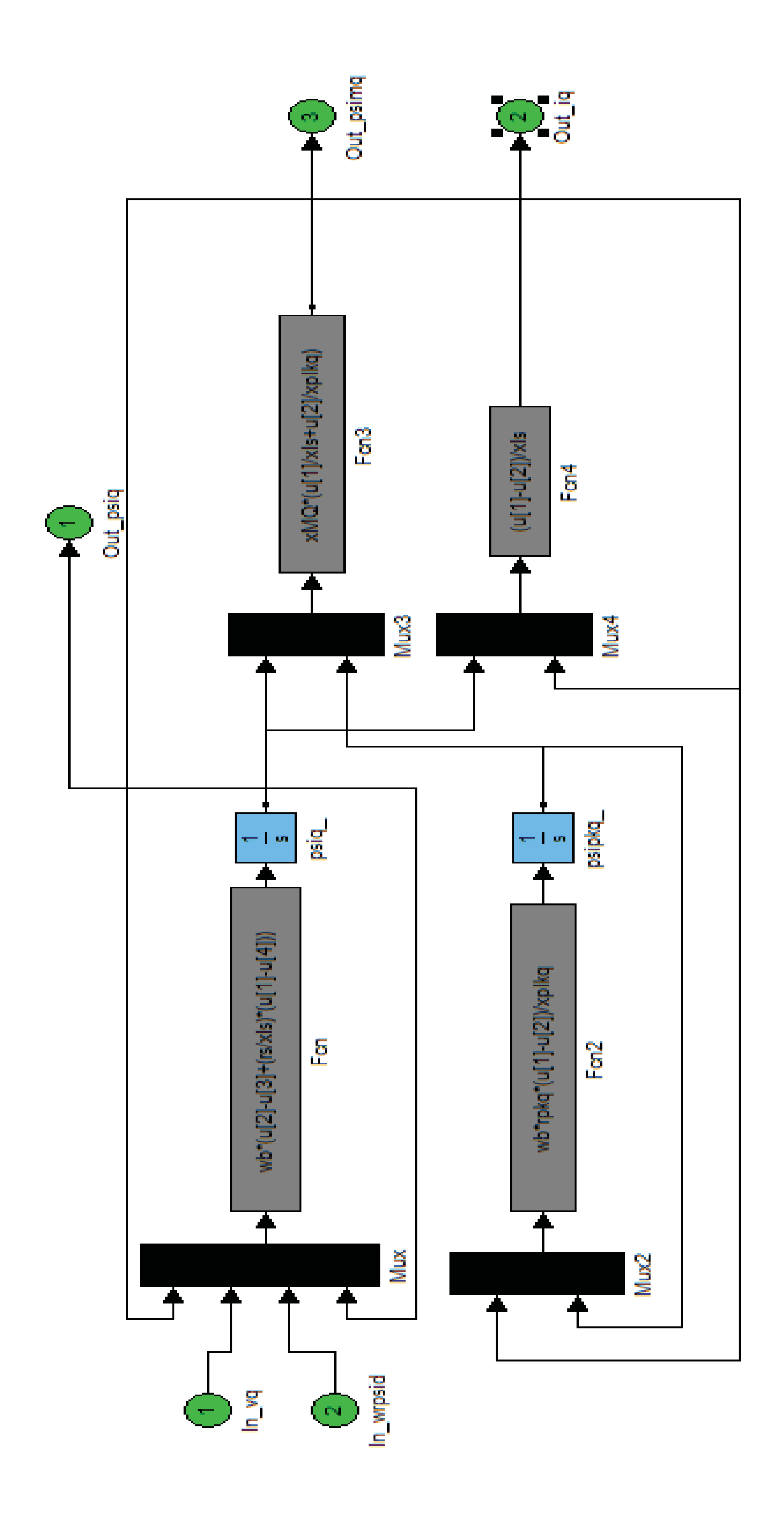
محتويات الصندوق ...١...



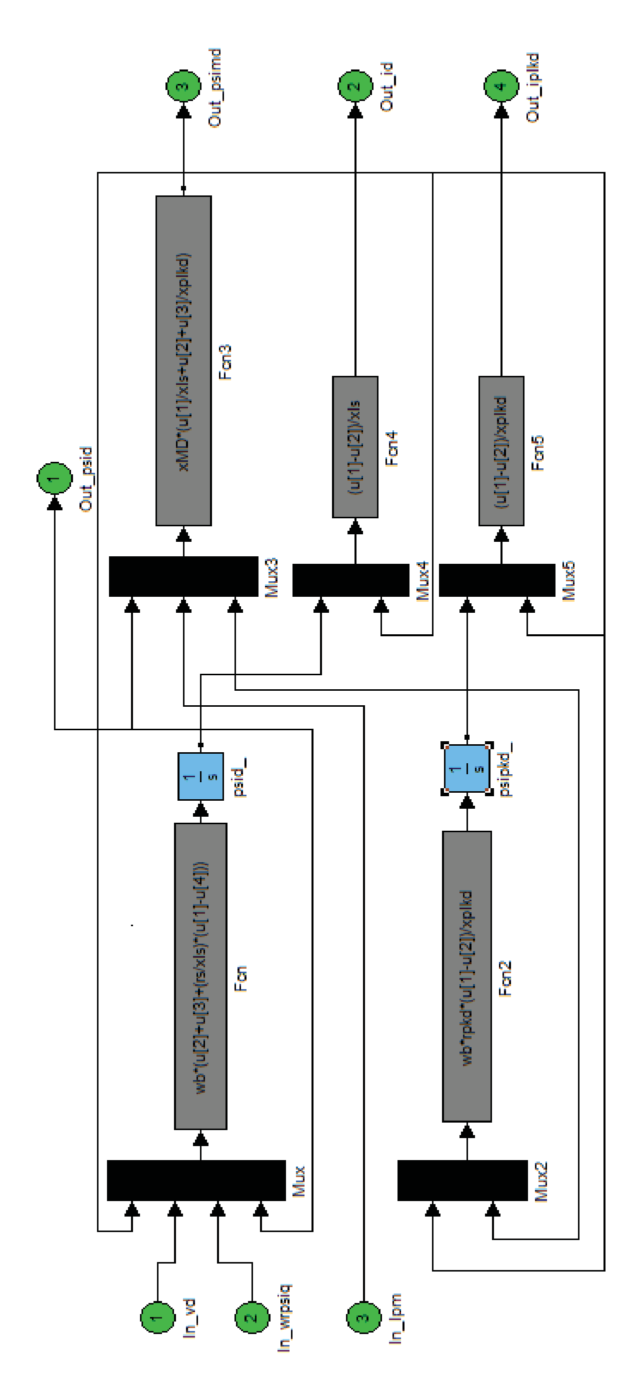
محتويات الصندوق ...٢...



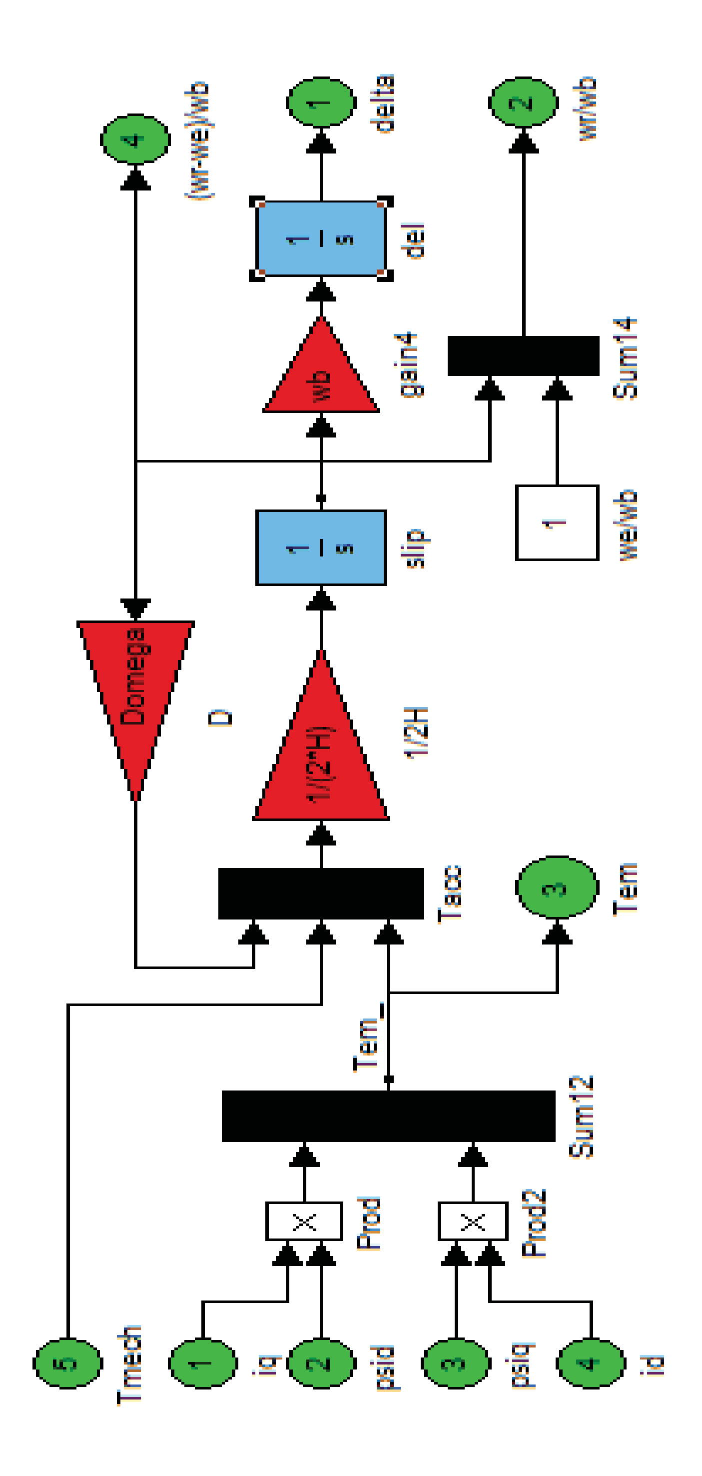
محتويات الصندوق ...a...



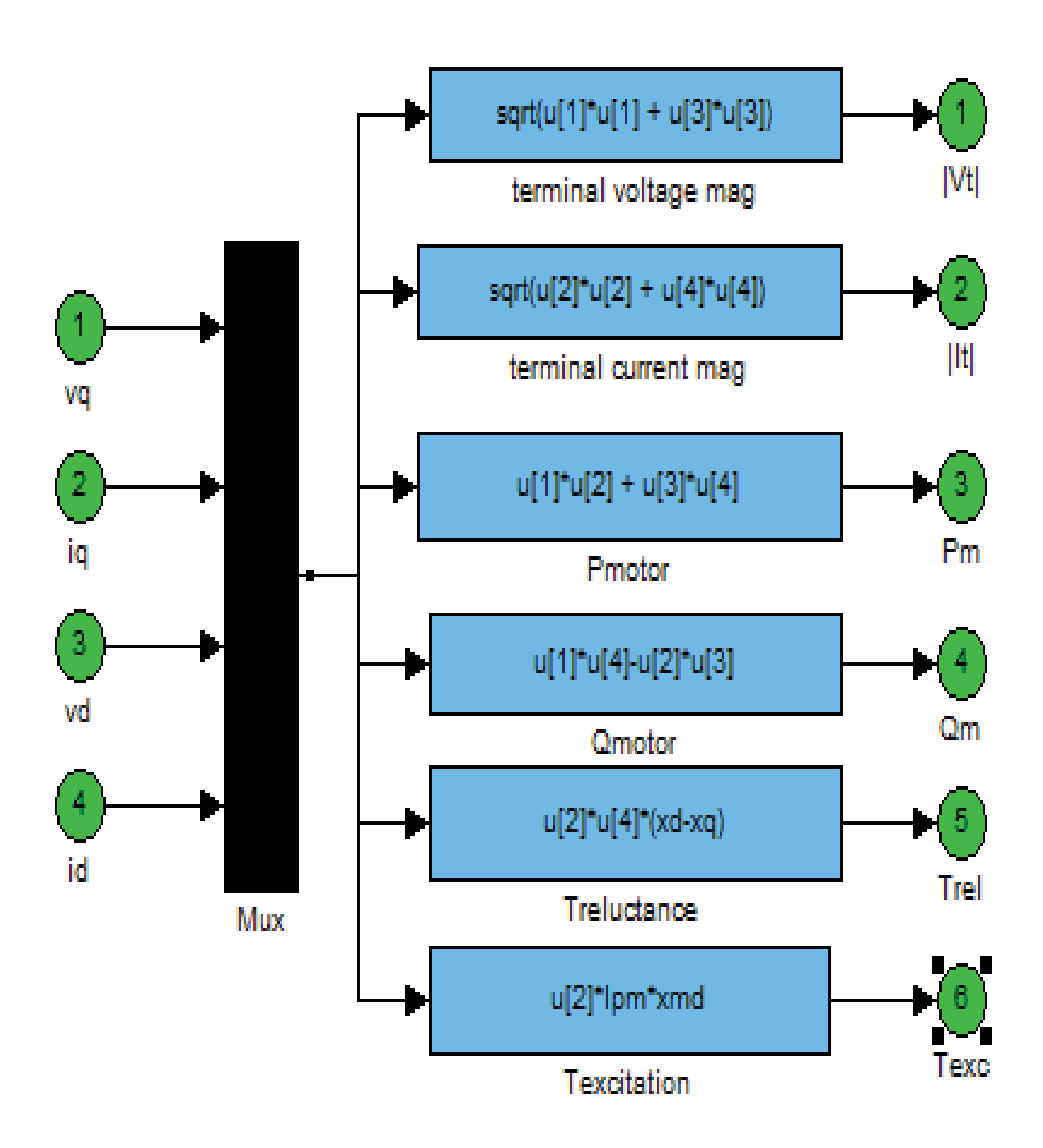
محتويات الصندوق ...b...



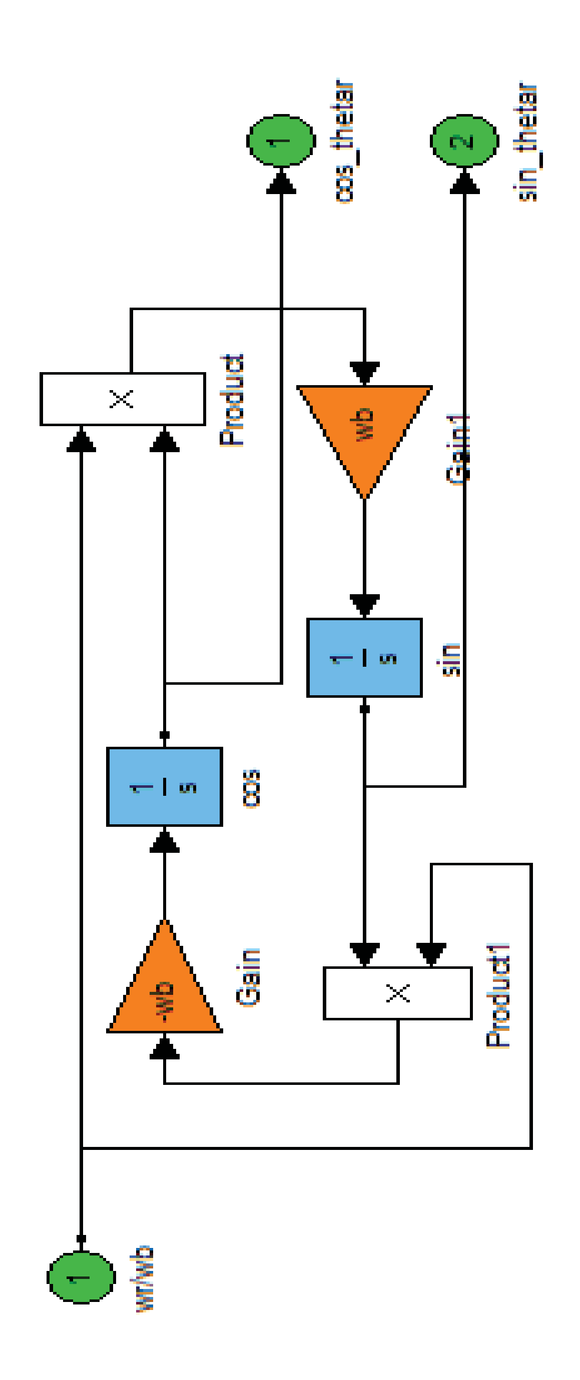
محتويات الصندوق ...c...

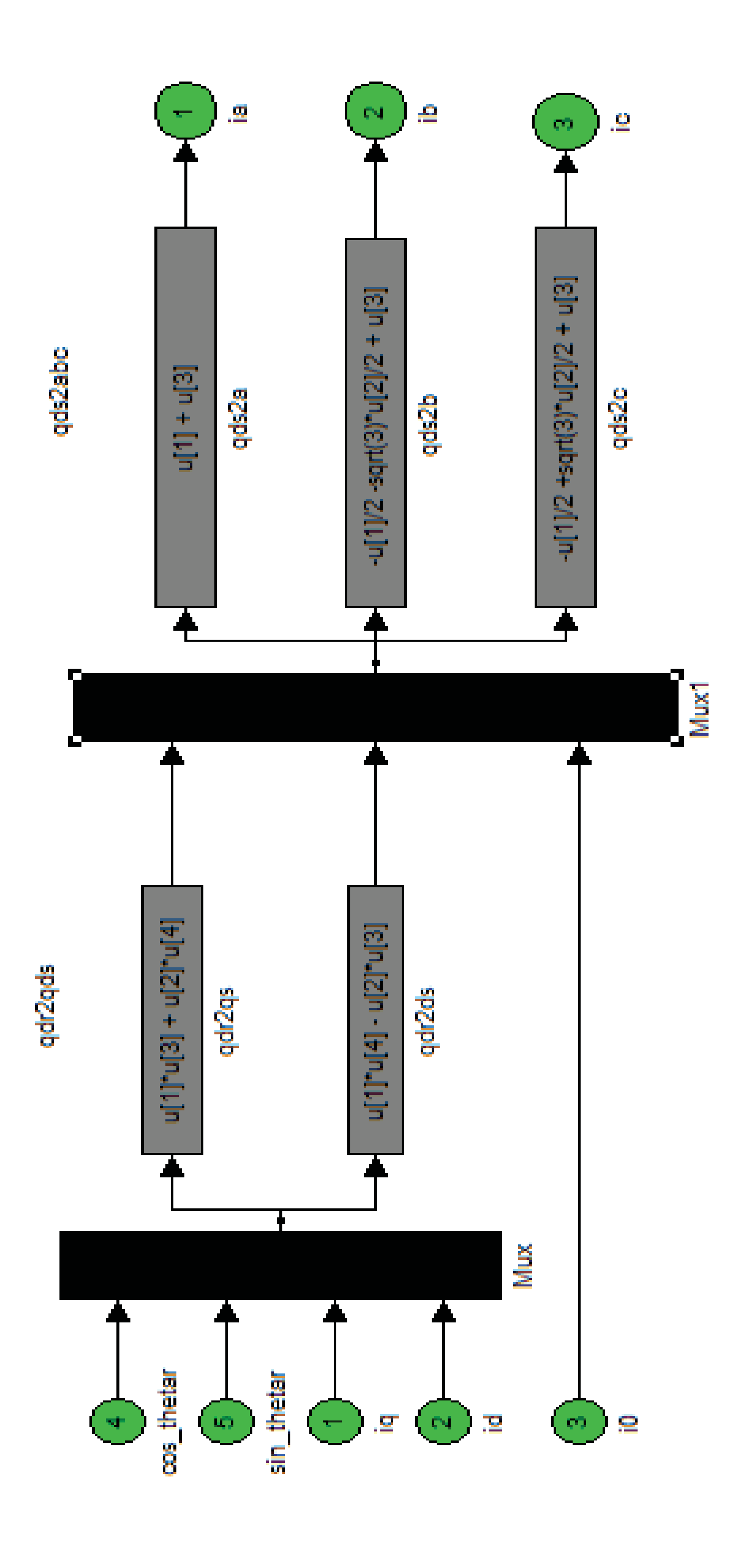


محتويات الصندوق ...3...



محتويات الصندوق ...4...





1
ia
2
ib
3
ic
qds2abc
u[1] + u[3]
qds2a
-u[1]/2 -sqrt(3)*u[2]/2 + u[3]
qds2b
-u[1]/2 +sqrt(3)*u[2]/2 + u[3]
qds2c
Mux1
qdr2qds
u[1]*u[3] + u[2]*u[4]
qdr2qs
u[1]*u[4] - u[2]*u[3]
qdr2ds
Mux
4
cos_thetar
5
sin_thetar
1
iq
2
id
3
i0

قمنا بمحاكاة لمحرك تزامني ذو مغناطيس دائم له البارمترات التالية :

$\mathrm{n} = 192\ \mathrm{r.p.m}$

F = 32 Hz

$n = \frac{120.F}{P} \rightarrow P = \frac{120.32}{192} = 20\ pole$

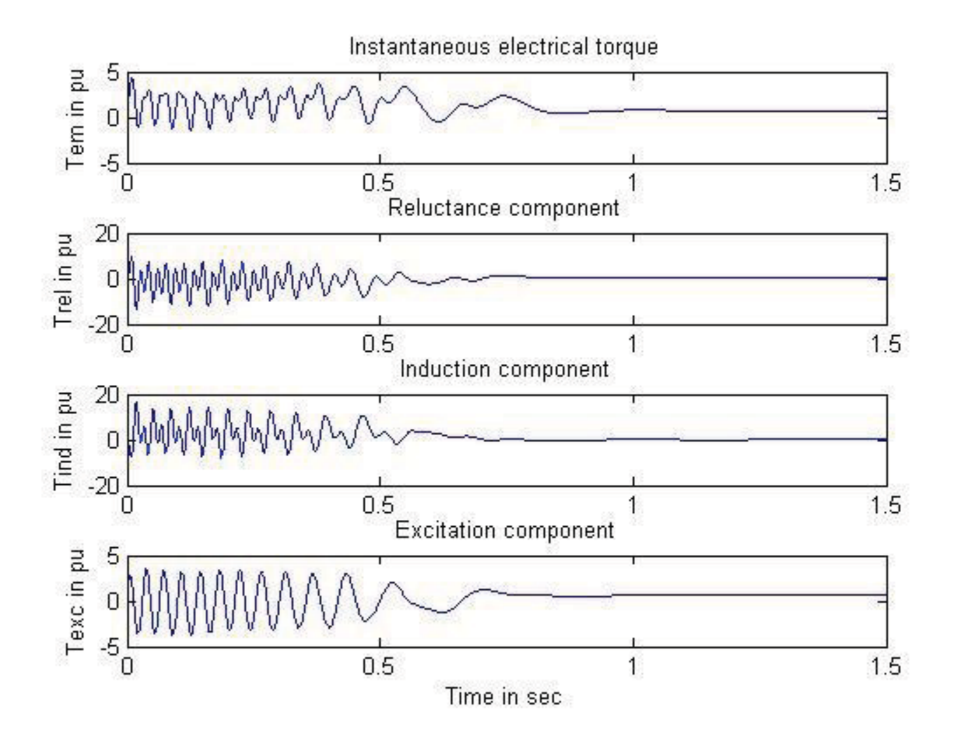
$V_t = 380\ V$

العزم الميكانيكي :

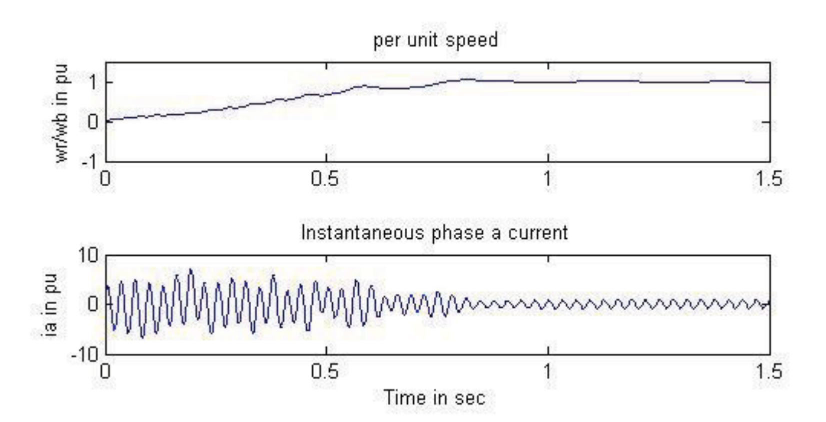
Tmech = -0.75 أي المحرك محمل بثلاث أرباع الحمل الاسمي له

النتائج :

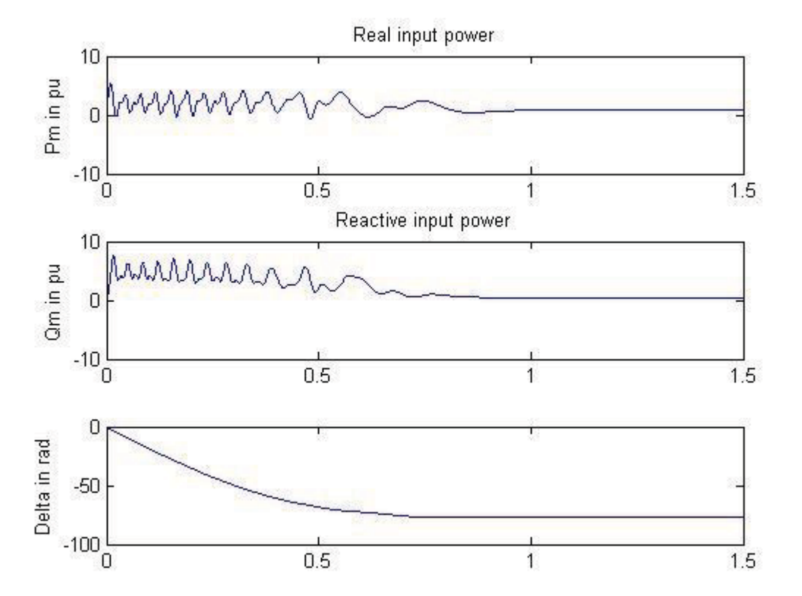
العزم (الكهرطيسي – المقاوم – التحريضي – التهييج)



السرعة - التيار



استطاعة الدخل (الحقيقية – الردية) ، زاوية الاستطاعة



المراجع العربية

- خطوات في احتراف الماتلاب ، المهندس موفق شما ، دار شعاع.
- تنفيذ وبرمجة واجهات المستخدم الرسومية GUI في الماتلاب ، الدكتور المهندس سميح يوسف العيسى ، طبعة 2007 دار شعاع.
- استخدام البيئة البرمجية (Simulink – SimPowerSys) في نمذجة ومحاكاة الدارات الكهربائية ، الدكتور مصطفى الحزوري و الدكتور علي الجازي ، جامعة دمشق ، طبعة 2007 – 2008.
- تصميم الواجهات الرسومية بالـ MATLAB– بن عيد.
- برمجة متحكمات AVR بلغة C – إعداد : اسماعيل الطرودي.
- الأردوينو ببساطة – م.عبد الله علي عبد الله.
- www.matlab4engineering.blogspot.com.
- قناة الماتلاب Youtube.

المراجع الاجنبية

- Electronics and Circuit Analysis using MATLAB , Ed. John Okyere Attia - Boca Raton : CRC Press LLC, 1999.
- Graphics and GUIs with MATLAB - Patrick Marchand and O. Thomas Holland : CRC Press company.
- Introduction to Simulink with Engineering Applications Steven T. Karris - Orchard Publications.
- Dynamic simulation of Electric Machinery using MATLAB – Chee mun Ong.
- Learning Basic Mechatronics concepts using the Arduino Board and MATLAB - Giampiero Campa, PhD.
- MATLAB By Example - Abhishek Kumar Gupta.
- Dynamic simulation of Electric Machinery using MATLAB.
- electronics and circuit analysis using MATLAB.
- ESSENTIAL MATLAB FOR ENGINEERS AND SCIENTISTS – Brian Hahn, Daniel T. Valentine.
- www.mathworks.com